Formation Debian GNU/Linux

Alexis de Lattre, Rémy Garrigue, Tanguy Ortolo, Adrien Grand, Loïc Alsfasser, Patrick Burri, et

18 juillet 2011

Table des matières

Résumé

Cette formation s'adresse aux personnes ayant déjà de bonnes connaissances informatiques, mais ne connaissant rien à GNU/Linux. Elle a pour but de les aider à installer à la main un système Debian GNU/Linux afin d'en découvrir le fonctionnement, l'utilisation et l'administration.

Si vous souhaitez simplement installer GNU/Linux sans vous poser de questions, cette formation ne vous est pas nécessaire, et je vous conseille d'installer un système Debian directement.

Pour nous contacter, vous pouvez :
- nous écrire à l'adresse formation-debian chez via.ecp.fr ;
- venir nous voir sur le salon Jabber debian-fr chez chat.jabberfr.org ;
- rejoindre notre canal IRC #formation-debian sur le réseau OFTC.

Ce document est disponible aux formats :
- HTML en ligne ou HTML zippé (3,5 Mio),
- PDF pour impression A4 (3,5 Mio),PDF zippé (2,2 Mio),
- EPUB pour lecture électronique (3,4 Mio).

Note

Si vous n'avez qu'un seul ordinateur, et que vous comptez suivre cette formation en l'imprimant, vous pouvez sauver quelques arbres en limitant votre impression au chapitre 22, à partir duquel vous pourrez suivre la formation en ligne.

AVERTISSEMENT

Les parties ne concernant pas l'installation de base (notamment la compilation du noyau) ont été déplacées sur notre wiki

Il existe un flux RSS signalant les dernières modifications apportées à ce document.

À propos de ce document

Objectif du document

Cette formation est destinée aux personnes qui ont déjà de bonnes connaissances en informatique mais qui ne connaissent encore rien à Linux. L'objectif est de les prendre en main pour passer en douceur de Windows à Debian GNU/Linux et de leur faire découvrir une partie des nombreuses possibilités de ce système d'exploitation.

Structure du document

Ce document est divisé en 5 parties :

1. I : je vous guiderai pas à pas pour l'installation brute du système d'exploitation ;
2. II : vous apprendrez les commandes et les outils de base et vous essaierez de faire marcher tous les périphériques de votre système ;
3. III : vous apprendrez à installer un bureau graphique et à vous servir des applications graphiques courantes sous Linux ;
4. IV : si vous avez un accès permanent à Internet ou si vous êtes connecté à un réseau local, vous découvrirez les qualités de Linux utilisé en tant que serveur au sein d'un réseau IP ;
5. V : vous découvrirez un certain nombre d'applications en console... et peut-être serez-vous séduit par les avantages de ce type d'applications !

Les utilisateurs qui débutent avec GNU/Linux seront plus particulièrement intéressés par les trois premières parties et certaines annexes « faciles ». Les utilisateurs qui ont déjà de l'expérience avec Linux, qui veulent explorer des utilisations plus avancées et qui ont déjà des bases en réseau pourront se plonger dans les parties 4 et 5 et les annexes plus difficiles.

Les parties 2, 3, 4 et 5 requièrent la connaissance des commandes de base d'Unix, qui sont rapidement abordées au Chapitre 14.

La formation doit absolument être suivie *dans l'ordre*. Elle a été conçue à l'origine pour les ordinateurs qui ont une connexion Internet par réseau local. Je l'ai ensuite adaptée pour ceux qui ont une connexion par modem classique, ADSL ou câble, voire pas de connexion du tout, mais il reste encore beaucoup à faire pour l'adapter à tous les modèles de modems ADSL USB.

Format du document

Pour maintenir cette formation, nous utilisons les formats et outils suivants :

Git pour stocker les sources de la formation et conserver ses anciennes versions

DocBook V5.0 comme format de document pour la rédaction ;

xsltproc pour produire le document HTML à l'aide d'une feuille de style XSL ;

Dblatex pour produire le document au format PDF.

Si vous souhaitez récupérer les sources de notre formation, vous aurez besoin du logiciel Git :

```
# aptitude install git
% git clone git://formation-debian.via.ecp.fr/formation- ↩
    debian.git
```

Si vous voulez compiler vous-même notre formation pour construire les versions HTML et PDF, vous aurez besoin des feuilles de style XSL DocBook et des logiciels xsltproc et Dblatex :

```
# aptitude install docbook-xsl-ns xsltproc dblatex
% cd formation-debian
% make html pdf
```

Contribuer au document !

Toutes les contributions sont les bienvenues ! Les contributions suivantes sont particulièrement appréciées :

– le signalement des fautes de frappe et d'orthographe,
– le signalement d'erreurs de syntaxe dans les commandes,
– le signalement de problèmes d'installation et de configuration survenus alors que toutes les instructions données dans ce document ont été scrupuleusement respectées,
– les informations sur l'installation sous Debian GNU/Linux de périphériques encore non documentés dans cette formation (certains modems ADSL USB, scanners, graveurs de DVD, etc.),
– des informations et des idées pour les points signalés par l'expression [TODO] dans le corps du document.

Merci d'envoyer vos contributions à l'adresse formation-debian chez via.ecp.fr.

ASTUCE

Nous acceptons toute correction ou remarque utile, mais si vous connaissez déjà bien les outils GNU/Linux, vous pouvez faciliter notre travail en nous proposant directement un *patch*. Pour cela, récupérez les sources de la formation, effectuez vos modifications, puis préparez votre *patch* avant de nous l'envoyer :

```
% git clone git://formation-debian.via.ecp.fr/formation ↩
    -debian.git
% vim formation-debian/src/partX/chapitre_à_modifier. ↩
    xml

% git commit -a --author='Robert Dupont <robert. ↩
    dupont@laposte.net>'
% git format-patch -1
```

Si vous désirez rédiger un nouveau chapitre ou une nouvelle annexe, merci de m'en parler au préalable via l'adresse ci-dessus pour qu'on en discute.

Je tiens à remercier les nombreux lecteurs qui ont contribué à ce document, par exemple en signalant des erreurs et en proposant des améliorations.

Première partie

Installation de Debian GNU/Linux

Cette première partie vous guidera pour installer la distribution *Debian GNU/Linux* sur un PC avec une architecture Intel. À la fin de cette première partie, le système d'exploitation sera installé... mais vous ne pourrez rien en faire ! Ce sont les parties suivantes qui vous apprendront à installer des programmes et à utiliser votre nouveau système. Je vous recommande de lire toute la première partie avant de débuter la procédure d'installation. Mais avant de commencer l'installation à proprement parler, je vais commencer par préciser ce que sont les logiciels libres, le projet GNU, Linux et Debian.

Chapitre 1

UNIX, Linux, GNU, logiciels libres… c'est quoi ?

1.1 Qu'est-ce qu'un logiciel *Libre* ?

La compilation

Un programme informatique, que ce soit un noyau ou un logiciel, est constitué de nombreuses lignes de code, écrites dans un langage de programmation (le langage C dans le cas du noyau Linux). Ce code n'est généralement pas utilisable en tant que tel. Il faut passer par la phase de *compilation* qui transforme le code source en programme exécutable, souvent appelé *binaire*. Il suffit d'avoir le *binaire* pour utiliser le programme ; on n'a pas besoin des sources du programme.

Il n'existe pas de moyen de remonter aux sources complètes du programme à partir du seul binaire. Quand on achète un logiciel (Microsoft Office par exemple) ou un système d'exploitation (Windows par exemple), on a un CD qui contient le binaire, mais pas les sources. Il est donc impossible de savoir comment le programme est conçu. Par conséquent, on ne peut pas modifier le programme. On peut seulement l'utiliser et éventuellement le copier à l'identique.

Les logiciels propriétaires et les logiciels Libres

Les logiciels propriétaires sont donc les logiciels dont une licence, souvent payante, ne donne qu'un droit limité d'utilisation. On n'a la plupart du temps accès qu'aux binaires de ces logiciels. Certains logiciels propriétaires sont gratuits, on les appelle alors des *freewares*.

Les logiciels *libres* sont les logiciels que l'ont peut librement utiliser, échanger, étudier et redistribuer. Cela implique que l'on ait accès à leur code source (d'où le terme équivalent *OpenSource*).

1.2 UNIX et les *Unix-like*

UNIX est un système d'exploitation, créé en 1969 par Ken Thompson et Dennis Ritchie. Sa conception a été particulièrement soignée, et apportait plusieurs innova-

tions, notamment le choix d'utiliser plusieurs outils simples et spécialisés, plutôt que des logiciels complexes à tout faire.

Ce système a été adapté par de nombreuses entreprises, qui ont ainsi développé des dérivés d'UNIX ou *Unices*, par exemple : XENIX de Microsoft, AIX d'IBM ou Solaris de Sun Microsystems.

D'autres systèmes d'exploitation ont ensuite été conçus en s'inspirant des principes d'UNIX, voire même en recherchant une compatibilité complète avec UNIX. Le nom UNIX étant une marque de certification, on parle alors plutôt d'*Unix-like*. Parmi ces systèmes, les plus connus sont GNU/Linux, les systèmes BSD, Minix ou encore Mac OS X.

1.3 Le projet GNU

Les logiciels libres

Scandalisé par les restrictions imposées par les logiciels propriétaires, Richard Stallman lance, en 1983, le projet GNU (logo Figure 1.1), qui a pour but de développer un système d'exploitation libre complet et inspiré d'UNIX, afin de contrer le développement croissant des logiciels propriétaires. L'histoire raconte que c'est une histoire de pilote d'imprimante qui lui a fait prendre conscience du danger de la logique propriétaire.



FIGURE 1.1 – Un GNU méditatif

Il fonde alors une association, la Free Software Foundation, (Figure 1.2) et écrit le GNU Manifesto, dans lequel il décrit les quatre libertés fondamentales que doit respecter un logiciel pour être qualifié de *logiciel libre* (*free software* en anglais) :

- la liberté d'exécution : tout le monde a le droit de lancer le programme, quel qu'en soit le but ;
- la liberté de modification : tout le monde a le droit d'étudier le programme et de le modifier, ce qui implique un accès au code source ;
- la liberté de redistribution : tout le monde a le droit de rediffuser le programme, gratuitement ou non ;
- la liberté d'amélioration : tout le monde a le droit de redistribuer une version modifiée du programme.



FIGURE 1.2 – La fondation pour le logiciel libre

La licence GPL

Pour donner un cadre juridique aux logiciels du projet GNU, il écrit une licence, la GNU General Public License alias GPL (il existe une traduction française non officielle). Cette licence reprend les quatre libertés fondamentales citées précédemment et impose pour la liberté d'amélioration que les versions modifiées d'un logiciel sous licence GPL ne peuvent être redistribuées que sous cette même licence. Richard Stallman invite alors tous les logiciels libres à adopter la licence GPL (le logo de cette licence est présenté en Figure 1.3).



FIGURE 1.3 – Le logo de la licence publique générale

Très tôt, Linus Torvalds adopte la licence GPL pour son noyau Linux. Aujourd'hui, de très nombreux logiciels libres sont distribués sous Licence GPL (VLC media player par exemple). Mais la licence GPL n'est pas la seule licence utilisée pour les logiciels libres. Par exemple, il existe aussi la licence BSD, qui diffère de la licence GPL par le fait qu'une version modifiée d'un logiciel sous Licence BSD peut être redistribuée sous une autre licence, même propriétaire.

1.4 Qu'est-ce que Linux ?

Un noyau

Linux désigne au sens strict un *noyau de système d'exploitation*.

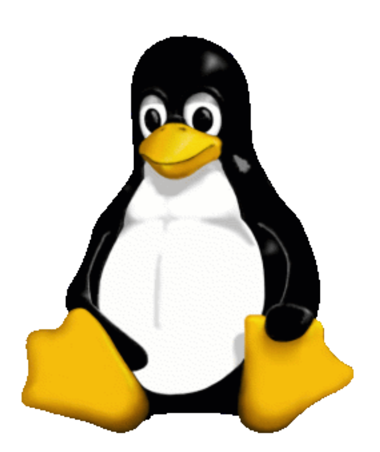


FIGURE 1.4 – Tux, la mascotte de Linux

Le noyau est la couche de base d'un système d'exploitation. C'est le noyau qui gère la mémoire, l'accès aux périphériques (disque dur, carte son, carte réseau, etc.), la circulation des données sur le bus, les droits d'accès, les multiples processus qui correspondent aux multiples tâches que l'ordinateur doit exécuter en même temps, etc.

Par contre, le noyau ne gère pas le mail, l'affichage des pages Web, ou encore le traitement du texte. Ce sont des *programmes* ou *applications* qui s'en chargent. Ces programmes viennent se greffer sur le noyau, et ils doivent être adaptés à celui-ci.

Ce noyau de système d'exploitation a l'originalité d'être multi-utilisateurs et multi-tâches et de fonctionner sur de nombreuses plates-formes (Intel, PowerPC, Sparc, etc.). Il est conforme à la norme POSIX et est distribué sous Licence GPL. Il a la réputation d'être fiable, stable et sécurisé. Son appartenance au monde du libre garantit une correction rapide des erreurs qui pourraient être découvertes.

Une distribution GNU/Linux

Comme nous venons de le voir, Linux ne se suffit pas à lui-même. Avec un simple noyau, on ne peut rien faire ! Le noyau Linux vient donc à l'intérieur de *distributions*.

Une distribution GNU/Linux, c'est un ensemble cohérent de plusieurs choses :

- un noyau Linux ;
- un ensemble de logiciels de base issus du projet GNU ;
- d'autres programmes, en libres (un navigateur Web, un lecteur de Mail, un serveur FTP, etc. . .), issus d'autres projets ;
- éventuellement, quelques logiciels propriétaires ;
- une méthode pour installer et désinstaller facilement ces programmes ;
- un programme d'installation du système d'exploitation.

Le noyau Linux ne se suffit donc pas à lui-même, mais on fait souvent un abus de langage en désignant par le terme *Linux* ce qui est en fait une *distribution GNU/Linux*.

Il existe de nombreuses distributions GNU/Linux : comme par exemple Red Hat, Mandriva, SUSE, Ubuntu ou Debian (celle que je vous propose d'installer).

La plupart des distributions sont gratuites, car constituées exclusivement de logiciels libres ou de programmes propriétaires gratuits. On peut donc télécharger les CD

librement sur Internet. On peut également acheter des boîtes contenant les CD dans le commerce. Les prix vont d'une dizaine d'euros pour couvrir les frais de presse des CD à plusieurs dizaines d'euros pour des produits comportant une documentation abondante et un support technique pour une certaine durée.

Un peu d'histoire

Linux naît en 1991 dans la chambre d'un étudiant finlandais, Linus Torvalds. Il développe un noyau en s'inspirant des principes du système d'exploitation UNIX. Son but initial est de s'amuser et d'apprendre les instructions Intel 386. Quand son noyau commence à marcher, il le met en libre téléchargement sur Internet en demandant aux gens de l'essayer et de lui dire ce qui ne marche pas chez eux.

De nombreuses personnes se montrent intéressées et l'aident à développer son noyau. Dès la version 0.12, il choisit de mettre Linux sous licence GPL. Quelques années plus tard, d'autres bénévoles commencent à créer des distributions Linux.

Aujourd'hui, le succès de Linux s'explique par la qualité technique du noyau, par la présence de nombreuses distributions Linux qui facilitent l'installation du système et des programmes, mais s'explique surtout par son appartenance au monde du Libre qui lui apporte une grande rapidité et qualité de développement. Le nombre d'utilisateurs de Linux est aujourd'hui estimé à plusieurs millions !

Si vous voulez en savoir plus sur l'histoire de Linux et de son père fondateur, Linus Torvalds, je vous conseille le livre qu'il a écrit intitulé *Just for fun - History of an accidental revolution*.

Chapitre 2

La distribution Debian



FIGURE 2.1 – Logo Debian

2.1 Pourquoi Debian ?

Nous avons choisi la distribution Debian pour plusieurs raisons :

- ses qualités techniques : Debian est réputée pour sa stabilité, pour son très bon système d'installation de mise à jour des composants logiciels et pour sa rapidité à réparer les failles de sécurité ;
- parce que c'est la distribution utilisée à VIA et VideoLAN, deux organisations dont Alexis, l'auteur original de cette formation, est membre ou ancien membre ;
- Debian GNU/Linux est utilisé par la plupart des fournisseurs d'accès à Internet, comme Free ;
- parce que c'est la première distribution GNU/Linux qu'Alexis a installée et utilisée, et qu'elle l'a toujours satisfait jusqu'à présent ;
- Debian est reconnu pour son sérieux et ses fortes prises de positions dans le monde libre. Debian garantit la liberté des logiciels qu'elle propose !

2.2 Ce qui différencie Debian des autres distributions

Au niveau de la philosophie

Debian est aujourd'hui la seule distribution majeure non commerciale. Debian est une organisation à but non lucratif constituée d'un millier de développeurs bénévoles répartis sur toute la planète (Figure 2.2). Elle est dirigée par un *project leader* élu par les développeurs. Les décisions se prennent au consensus ou par vote.

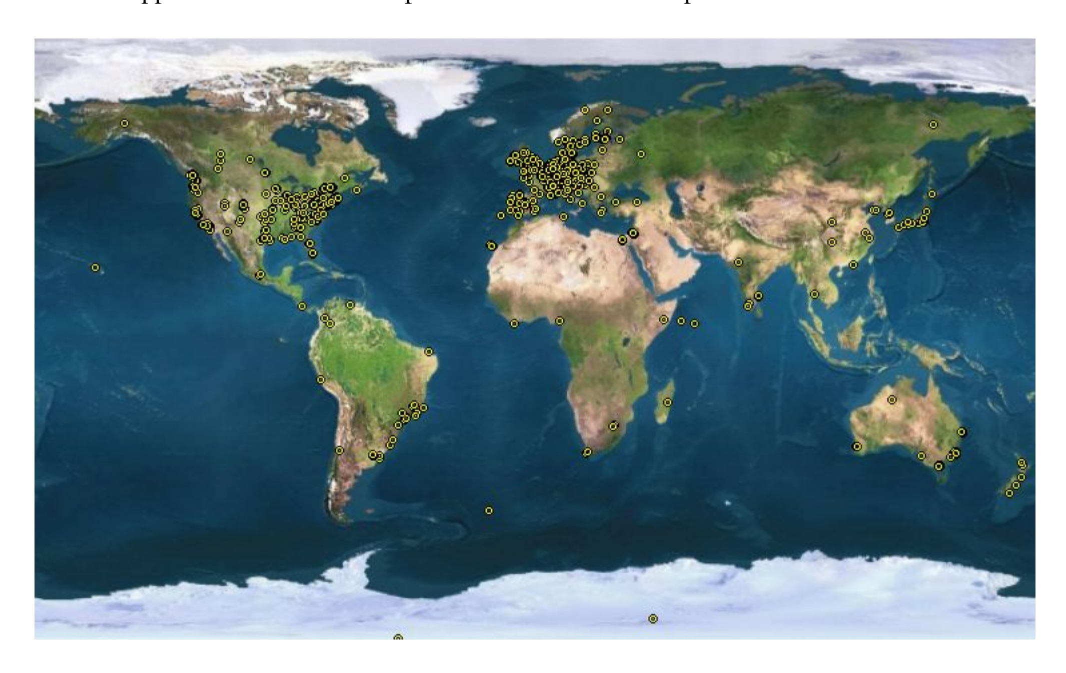


FIGURE 2.2 – Carte des développeurs Debian

Les autres distributions GNU/Linux majeures sont des sociétés commerciales, ce qui ne les empêche pas de produire des logiciels libres !

Debian se distingue aussi par son attachement très fort à la philosophie du logiciel libre. Cet attachement est forgé dans son Contrat Social et dans Les principes du logiciel libre selon Debian ; qui sont deux textes relativement courts que je vous invite à lire.

Au niveau technique

Qu'est-ce qu'un *paquet* ?

Un *paquet* est un logiciel ou une partie d'un logiciel, qui a été préparé dans un format spécial, afin de faciliter sa recherche, la consultation d'informations à son sujet, ainsi que son installation et sa désintallation. Ce paquet prend la forme d'un fichier avec un nom particulier : `nom-du-logiciel_numéro-de-version_nom-de-l'architecture.deb` (par exemple le fichier `apache_1.3.24_i386.deb` contient la version 1.3.24 du programme Apache pour processeurs Intel). Ce fichier contient les binaires du programme ainsi qu'un certain nombre d'en-têtes. Ces en-têtes contiennent :

– le nom du paquet, son numéro de version, l'architecture pour laquelle il a été compilé, et la catégorie à laquelle il appartient (client réseau, jeu, utilitaire. . .) ;
– le nom du développeur Debian qui s'en occupe et son adresse e-mail ;
– une description du logiciel qu'il contient ;
– le nom et la version des autres paquets dont il dépend ainsi que des autres paquets avec lesquels il entre en conflit.

Le système de gestion des paquets

Le système de gestion des paquets de Debian est très performant et très facile à utiliser. Grâce à lui, les logiciels s'installent, se retirent et peuvent être mis à jour très facilement. Vous le découvrirez dans la deuxième partie de cette formation.

La stabilité

Debian GNU/Linux est réputé pour être un système d'exploitation très stable. Avant chaque nouvelle version, le système est longuement testé et il ne sort qu'une fois que tous les bogues connus ont été corrigés. Debian s'est doté d'un Bug Tracking System (BTS) très performant et très pratique qui permet aux développeurs d'avoir un retour d'expérience instructif des utilisateurs, ce qui les aide à corriger les bogues rapidement.

La procédure d'installation

Même si ce n'est pas la procédure d'installation la plus agréable à l'œil de toutes les distributions Linux, elle n'en reste pas moins une procédure rapide et efficace, qui permet de bien maîtriser l'installation de son système d'exploitation.

Les architectures

Debian GNU/Linux est disponible sous 12 architectures, dont Intel 32 et 64 bits, PowerPC (les anciens Macintosh) et Sparc (les stations Sun).

En savoir plus...

Pour en savoir plus sur Debian (ses textes fondateurs, son histoire, son organisation et son fonctionnement technique), je vous invite à lire les transparents de la conférence sur Debian donnée par Samuel Hocevar le 24 avril 2002 et intitulée *« Debian, what your mom would use if it was 20 times easier »* !

2.3 Les différentes versions de Debian

Il existe trois versions de Debian activement maintenues :
– une version *officielle stable*, nommée *Squeeze*, numérotée *6.0* ;
– une version *testing*, nommée *Wheezy* qui est en fait la future version stable ;
– une version *unstable*, appelée *Sid*, pour *still in development* (encore en développement), destinée à tester les nouveaux paquets avant qu'ils ne soient placés dans la *testing*.

Chaque version a son utilité. Les avantages et inconvénients de chaque version sont présentés dans le tableau ci-dessous :

Nom	type	avantages	inconvénients	utilisation
Squeeze	stable	stabilité et mises à jour de sécurité suivies	aucune évolution des paquets	pour les serveurs et les utilisateurs « normaux »
Wheezy	testing	paquets plus nombreux et plus récents	quelques bogues, évolution permanente	sert à élaborer la future version stable de Debian, adaptée pour l'utilisation courante d'utilisateurs expérimentés
Sid	unstable	paquets encore plus nombreux et les plus récents possible	logiciels peu ou pas testés, évolution permanente	accueille les nouvelles versions des logiciels avant qu'ils ne soient testés, adaptée utilisateurs expérimentés prêts à supporter des bogues potentiellement gênants pour participer à l'élaboration de Debian

TABLE 2.1 – Les trois versions de Debian

Je vous propose dans cette formation de commencer par installer une *Squeeze*. Il faut savoir que vous pouvez passer facilement d'une version donnée à une version supérieure, mais l'inverse est plus difficile. Donc si vous installez une Squeeze, vous pourrez passer facilement en Wheezy ou en Sid ; mais vous ne pourrez que difficilement revenir en Squeeze ensuite.

2.4 L'histoire

La première version de Debian, la 0.01 est sortie en 1993. Puis les versions s'enchaînent, avec des noms inspirés du film Toy Story (v1.1 alias *Buzz* en 1996 ; v1.2 alias *Rex* en 1996 ; v1.3 alias *Bo* en 1997 ; v2.0 alias *Hamm* en 1998 ; v2.1 alias *Slink* en 1999 ; v2.2 alias *Potato* en 2000 ; v3.0 alias *Woody* en 2002, v3.1 alias *Sarge* en 2005), v4.0 alias *Etch* en 2007, et enfin Lenny, v5.0 en 2009.

Alors que Debian n'était composée que de quelques programmeurs à ses débuts, l'organisation compte aujourd'hui un millier de développeurs répartis sur toute la planète et qui s'occupent de quelques 36103 paquets ! Pour en savoir plus sur l'histoire de Debian, je vous invite à lire le document A brief history of Debian.

Chapitre 3

Motivation et matériel requis

3.1 Motivation

La découverte de GNU/Linux demande beaucoup de motivation. Il faut tout réapprendre de zéro, surtout pour ceux qui ne sont pas familiers avec le monde Unix. Cela signifie de longues heures d'apprentissage avec son lot d'essais infructueux et de déceptions. Une certaine dose de ténacité est donc nécessaire !

Après quelques semaines, vous aurez (j'espère !) la satisfaction de maîtriser un système d'exploitation fiable et puissant, qui vous permettra de réaliser des choses que vous ne pouviez pas faire auparavant. Si en plus vous aimez ce qui touche au réseau, alors vous serez probablement comblé : GNU/Linux a d'abord été un système d'exploitation pour les serveurs qui doivent rester branchés au réseau 24h/24, avant d'acquérir, plus récemment, des fonctions multimédia pour le grand public.

3.2 Le matériel requis

La configuration requise

Cette formation ne couvre que l'installation de Debian GNU/Linux sur un PC avec une architecture Intel (processeurs Intel ou AMD). Linux est un système d'exploitation peu gourmand en ressources. Pour une utilisation bureautique normale, un ordinateur avec 256 Mo de mémoire vive et 5 Go d'espace libre sur le disque dur est nécessaire. Si vous avez 512 Mo de mémoire ou plus, les applications graphiques seront plus agréables à utiliser car plus rapides.

Choix de la méthode d'installation

Dans cette formation, j'explique 2 méthodes d'installation différentes :

- méthode *netinstall*, pour ceux qui ont une connexion filaire haut-débit à Internet ;
- méthode *53 CD / 8 DVD / X Blu-ray*, pour ceux qui ont une connexion bas débit à Internet, pas de connexion du tout, ou qui ne disposent que d'une connexion sans fil peu propice à une installation par réseau !

Matériel requis selon la méthode d'installation

Méthode netinstall

Ce mode d'installation utilise un support contenant un système d'installation minimal qui ira directement télécharger tous les logiciels que vous voudrez installer. Vous aurez besoin d'un disque inscriptible sur lequel graver le système d'installation, ou, si votre ordinateur est capable de démarrer dessus, d'une clef USB d'au moins 256 Mo.

Méthode 53 CD / 8 DVD / X Blu-ray

Il faut que vous vous procuriez les 53 CD, les 8 DVD ou les X disques Blu-ray de la Debian version *6.0*. Il est possible d'acheter ces disques auprès d'un vendeur, ou de les télécharger par HTTP, FTP ou BitTorrent ; ces différents moyens sont détaillés sur le site officiel de Debian.

Note

Les logiciels proposés par Debian sont distribués sur les disques d'installation selon leur popularité. Ainsi, le premier CD seul contient de quoi installer un bureau graphique minimal. Le premier DVD ou Blu-ray, ou les quelques premiers CD devraient suffire pour pouvoir installer un bon nombre de logiciels courants.

Dans tous les cas

Si vous comptez faire cohabiter Windows et Debian sur le même ordinateur, vous aurez également besoin d'un tout petit peu de place sur votre partition Windows ; si vous comptez installer Debian seul, vous aurez besoin d'une disquette, d'une clef USB ou d'un CD vierge.

Le manuel officiel d'installation

Les instructions de cette formation sont normalement suffisantes, mais ceux qui ont une configuration ou des besoins particuliers trouveront toutes les informations sur l'installation dans le manuel d'installation officiel.

Chapitre 4

Les préliminaires

4.1 Réfléchir au partitionnement du disque dur

Quelle place allouer à GNU/Linux ?

Ce qu'il faut tout d'abord savoir, si vous voulez faire cohabiter Linux et Windows sur le même ordinateur, c'est que vous pouvez avoir accès :

- à vos partitions Windows depuis GNU/Linux en lecture et écriture,
- à vos partitions GNU/Linux depuis Windows en lecture et écriture.

Il vous faudra de toute façon plusieurs partitions Linux, en prévoyant un minimum de :

- 300 Mo environ pour un petit système sans serveur graphique ;
- 1 Go pour un petit système avec un serveur graphique et quelques applications graphiques ;
- 5 Go pour un système complet avec un serveur graphique et de nombreuses applications graphiques et des outils de développement avancés.

N'oubliez pas de compter en plus les fichiers personnels que vous voudrez stocker sur vos partitions GNU/Linux !

Quelle organisation du disque dur ?

La théorie des partitions

Chaque disque dur peut contenir quatre *partitions primaires* au maximum. Si vous voulez plus de quatre partitions, il va falloir transformer une des partitions primaires en *partition étendue*, aussi appelée *primaire étendue*. Dans cette partition étendue, vous pouvez créer un nombre illimité de *lecteurs logiques*, qui formeront autant de partitions (exemple Figure 4.1).

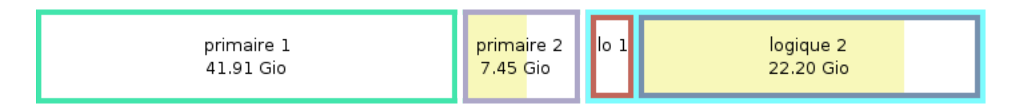


FIGURE 4.1 – Exemple de partitionnement

Conseils pour faire cohabiter Windows et Linux

Je vais donner quelques conseils pour une cohabitation de Windows et GNU/Linux qui sont les deux seuls systèmes d'exploitation que je connais bien.

Pour faire cohabiter GNU/Linux et Windows, vous pouvez adopter l'organisation suivante pour votre disque dur :

1. d'abord une partition primaire pour Windows ;
2. ensuite une grande partition étendue découpée en deux lecteurs logiques (ou plus) pour Linux.

Ne cherchez pas encore à repartitionner votre disque dur : l'étape de partitionnement proprement dite aura lieu lors de l'installation de Debian, au Chapitre 11. Nous aurons alors le temps de découvrir plus en détail l'utilisation du partitionnement sous GNU/Linux.

4.2 Sauvegarder ses données

Il est très fortement recommandé de sauvegarder toutes les données importantes (fichiers perso, courrier, etc.) se trouvant sur l'ordinateur sur lequel se fera l'installation : quand on installe un nouveau système et qu'on touche au partitionnement du disque dur, une mauvaise manipulation (ou une coupure de courant) est toujours possible !

4.3 Informations à obtenir avant de commencer

Type de processeur

Debian GNU/Linux est capable de fonctionner sur de 12 architectures matérielles. En ce qui nous concerne :

- les PC de plus de trois ans, équipés de processeurs Intel Pentium, Centrino ou AMD Athlon, de type *i386* ;
- les PC et MacIntosh récents, équipés de processeurs Intel Core, Centrino ou AMD 64 bits, de type *amd64*.

Pour des raisons de compatibilité, les processeurs de type *amd64* sont en fait également capable de fonctionner comme des *i386*. On peut donc y installer aussi bien Debian amd64 que Debian i386. Aujourd'hui, l'ensemble des logiciels fournis par Debian peut fonctionner en mode 64 bits. En revanche, les logiciels propriétaires sont encore très en retard dans ce domaine : le plugin Adobe Flash n'est par exemple toujours pas disponible en version 64 bits.

En définitive, si vous ne savez de quel type de processeur vous disposez, ou si vous comptez visiter de nombreux sites Web en Flash, choisissez (à regret) Debian i386.

Note

Plusieurs alternatives libres sont disponibles pour amd64 et peuvent remplacer le plugin Adobe Flash. Elles permettent sans problème de visiter des sites en Flash pas trop compliqués, ainsi que de regarder des vidéos sur les sites comme YouTube. En revanche, la plupart des jeux Flash y sont inutilisables.

Si vous êtes connecté à un réseau local

Si l'ordinateur est connecté à un réseau local relié à Internet, renseignez-vous pour savoir s'il y a un serveur DHCP sur le réseau local (cas de la majorité des réseaux et en particulier du réseau VIA) :

- si un serveur DHCP est présent, vous n'avez rien à faire, la configuration réseau se fera automatiquement lors de la procédure d'installation,
- s'il n'y a pas de serveur DHCP, il faut que vous connaissiez vos paramètres réseau (adresse IP, masque de sous-réseau, passerelle, nom de domaine, adresse des serveurs DNS) avant de commencer l'installation. Vous pouvez les obtenir avec la commande **ipconfig /all** sous Windows.

AVERTISSEMENT

Si vous êtes habituellement connecté à un réseau sans fil, je vous conseille, le temps de l'installation, d'utiliser une connexion Ethernet filaire pour pouvoir effectuer une installation par réseau (*netinstall*).

Si vous avez un modem USB

Si vous avez un modem ADSL ou câble USB, il faut que vous connaissiez précisément le modèle de votre modem.

Si vous avez un modem Ethernet

Dans le cas d'un modem Ethernet, vous devez déterminer son type :

- s'il s'agit d'une *Box ou d'un modem routeur, il gère lui-même la connexion : vous êtes en fait dans le cas d'un réseau local avec serveur DHCP ;
- s'il s'agit d'un modem plus ancien, vous devrez probablement régler votre ordinateur pour vous connecter en PPP over Ethernet (PPPoE).

Chapitre 5

Création du support d'installation

5.1 Création du médium de démarrage

Méthode *53 CD / 8 DVD*

Le médium de démarrage est entre vos mains : le CD ou DVD numéro 1 est démarrable et va servir pour lancer l'installation.

Méthode CD *netinstall*

Télécharger l'image du CD

Vous allez télécharger l'image du CD « netinstall » de Debian version *6.0* (cette image pèse environ 190 Mo). Cette image est démarrable et contient le minimum nécessaire pour mener la procédure d'installation jusqu'à son terme. Selon votre type de processeur (i386 ou amd64, cf. Chapitre 4), téléchargez l'image i386 ou amd64.

Note

Ces images ne pèsent que 150 Mo, ce qui permet de les graver sur des petits disques de 8 cm de diamètre. Debian propose également des images de CD de 40 Mo, destinées à être gravées sur des CD rectangulaires au format carte de visite, dont la surface utile est plus faible. Il est toutefois inutile d'utiliser ces images pour économiser du volume de téléchargement, étant donné qu'elles ne contiennent pas un installateur complet, et que leur première action est donc d'aller télécharger celui-ci.

Vérifier l'intégrité de l'image du CD

Pour vérifier qu'un fichier est bien le vrai fichier original, on calcule la somme de contrôle SHA-512 du fichier, et on la compare à la somme SHA-512 officielle. Deux fichiers ayant la même somme SHA-512 sont normalement identiques.

Note

Pour calculer la somme SHA-512 d'un fichier sous Windows, vous pouvez utiliser l'utilitaire Fsum Frontend avec le fichier de sommes de contrôle correspondantà votre architecture : i386 ou amd64.

Et MD5 ?

Vous avez peut-être l'habitude de vérifier vos téléchargement à l'aide de leur somme MD5. La fonction MD5 a cependant été compromise, et ne permet donc plus de garantir qu'un fichier n'a pas été modifié. Il faut donc utiliser les fonctions de la famille SHA à la place de MD5.

Graver le CD

Gravez l'image du CD en utilisant votre logiciel de gravure favori. Cette page explique comment faire, selon le logiciel que vous utilisez.

Méthode clef USB *netinstall*

Télécharger l'image de la clef

L'installateur Debian a été conçu pour que les images de CD puissent être utilisées sans modification sur des clefs USB. Comme pour la méthode CD, vous allez donc télécharger l'image du CD « netinstall ». Selon votre type de processeur ((i386 ou amd64, cf. Chapitre 4), téléchargez l'image i386 ou amd64.

Vérifiez ensuite l'intégrité de l'image comme indiqué pour la méthode CD.

Écrire la clef USB

L'image de la clef USB doit être écrite directement sur la clef, et non placée comme un fichier. Sous Windows, vous pouvez utiliser l'utilitaire Win32 Disk Imager : sélectionnez le fichier de l'image, puis la lettre correspondant à votre clef USB, puis lancez l'écriture.

5.2 Les fichiers supplémentaires

Téléchargez l'archive contenant nos fichiers de configuration, et placez-la sur une clef USB : vous en aurez besoin après avoir effectué l'installation du système Debian.

Note

Cette archive est au format *tarball*, qui est le plus courant sous GNU/Linux et les autres systèmes Unix-like. Si vous souhaitez y jeter un œil sous Windows, vous pouvez l'ouvrir avec un logiciel comme 7-Zip.

Téléchargez également l'archive contenant les micro-programmes nécessaires pour utiliser certains périphériques. Décompressez-la et placez les fichiers qu'elle contient dans un répertoire nommé `/firmware` sur une clef USB.

Note

Certains périphériques, notamment les cartes réseau sans fil et certaines cartes réseau filaires, nécessitent pour fonctionner des micro-programmes propriétaires. Ces micro-programmes ne sont pas destinés à être exécutés sur le processeur central de l'ordinateur mais sur celui du périphérique concerné. Il s'agit néanmoins de code propriétaire, qui ne peut pas être distribué normalement avec l'installateur Debian, d'où cette étape.

Chapitre 6

Préparation du disque dur

Je suppose que votre disque dur contient simplement une seule partition primaire dédiée à Windows au format FAT ou NTFS qui couvre tout le disque dur.

6.1 Démarche

Pour savoir si votre partition est formatée en FAT ou NTFS, allez dans le Poste de Travail, faites un clic droit sur le lecteur `C:` : c'est écrit dans l'onglet *Général*.

Lorsque Windows doit écrire un fichier, il le fait à la suite du dernier fichier écrit. Mais quand vous supprimez des fichiers, Windows ne remplit pas systématiquement les trous de votre partition. Si bien que rapidement, votre partition ressemble au gruyère de la Figure 6.1.



FIGURE 6.1 – Disque fragmenté

La première opération consiste donc à faire une *défragmentation*, c'est-à-dire défragmenter vos fichiers et mettre de l'ordre dans la partition, de sorte qu'il ne reste qu'un bloc de données compact en début de disque, comme sur la Figure 6.2.

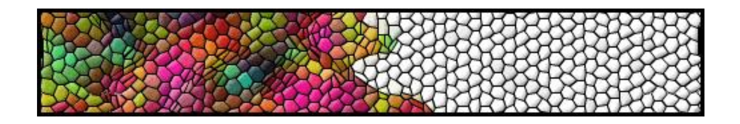


FIGURE 6.2 – Disque défragmenté

6.2 Défragmenter

C'est très simple : sous Windows, allez dans le poste de travail et faites un clic droit sur votre disque dur (lecteur *C:* normalement). Sélectionnez *Propriétés*, allez dans l'onglet *Outils* et cliquez sur *Défragmenter maintenant*.

C'est un peu long, je vous l'accorde, mais d'une part ça fait du bien à votre partition, et d'autre part c'est nécessaire alors patience !

AVERTISSEMENT

Avant d'aller plus loin, j'espère que vous avez bien fait une sauvegarde des données importantes de votre disque dur comme je vous l'ai conseillé.

6.3 Notez l'espace disponible

Regardez quelle est la quantité d'espace disponible sur votre partition Windows, et décidez en conséquence (et en fonction de vos besoins !) la taille de l'espace disque que vous allez allouer à Debian. Sachez qu'un système Debian installé avec beaucoup de logiciels occupe typiquement *5 Go*, sans compter les fichiers de l'utilisateur. Vous lisez bien, 5 Go, pas 15 ni 50 !

Chapitre 7

Débuter l'installation

Vous allez enfin commencer la procédure d'installation de Debian !

7.1 Démarrer sur le disque d'installation

Régler la séquence de démarrage

Redémarrez votre ordinateur et entrez dans le BIOS en appuyant sur une touche au démarrage (généralement **Suppr** ou **F1**).

Naviguez dans les menus du BIOS jusqu'à la page qui permet de changer la séquence de démarrage. Selon votre méthode d'installation, assurez-vous que le lecteur optique désigné par *CD-ROM*, ou la clef USB, y intervient avant le disque dur désigné par *C*, *Hard Drive* ou *HDD-0*.

Quittez le BIOS en sauvegardant les changements (**F10**).

C'est parti...

Insérez le CD netinstall, le premier disque ou la clef USB et redémarrez votre ordinateur. Quand vous voyez l'écran correspondant à la Figure 7.1,

- si vous utilisez un modem PPPoE, éditez l'entrée *Install* en appuyant sur la touche **Tab**, puis tapez : **`modules=ppp-udeb`** (attention, votre clavier fonctionne alors comme un clavier anglais, vous devrez donc appuyer sur les touches correspondant à **`,odules=ppp)udeb`**), puis appuyez sur **Entrée**.
- dans tous les autres cas, validez directement pour lancer l'installation en mode semi-graphique.

Note

Vous pouvez également choisir d'utiliser l'installateur graphique, qui vous proposera *exactement* les mêmes menus, mais avec un pointeur de souris et un aspect plus soigné, comme le montre la Figure 7.2. Personnellement, je trouve qu'il est un peu moins réactif que l'installateur *semi-graphique* classique.

Le système devrait alors démarrer sans problème.



FIGURE 7.1 – Le premier écran



FIGURE 7.2 – L'installateur en mode graphique. . .

Ça marche ?

Si vous avez un ordinateur portable et que ce dernier redémarre brutalement pendant la procédure d'installation, éditez l'entrée *Install* en appuyant sur la touche **Tab**, puis tapez **`noacpi noapic nolapic`** à l'écran d'accueil au lieu de simplement valider, ce qui désactivera l'ACPI (un système d'économie d'énergie qui est bogué sur certaines cartes mères).

Mais dans l'immense majorité des cas, le démarrage se passe sans problème et vous arrivez alors dans l'interface bleu-blanc-rouge qui va vous accompagner tout au long de cette installation.

ASTUCE

Pendant toute la durée de l'installation, l'installateur écrit des rapports concernant chacune de ses actions. Si vous souhaitez voir de quoi il s'agit, vous pouvez les consulter en appuyant sur Alt-F4. Vous pouvez ensuite retourner sur l'interface d'installation en appuyant sur Alt-F1.

7.2 Choix de la langue puis du clavier

Choisissez la langue *French / Français* et ensuite votre pays.

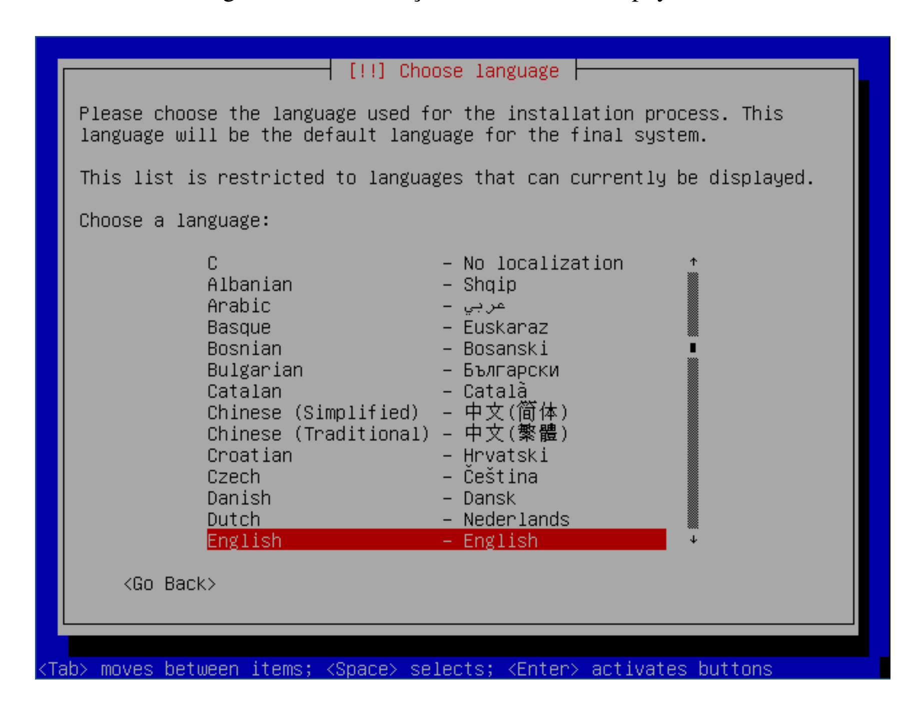


FIGURE 7.3 – Choix de la langue

Ensuite, il faut choisir la configuration de votre clavier. Une bonne configuration du clavier est indispensable dans la mesure où vous aurez à saisir vos mots de passe

qui n'apparaîtront jamais en clair à l'écran. Si vous avez un clavier français, choisissez le clavier proposé par défaut *Français (fr-latin9)*.

```
[!] Choisir la disposition du clavier

Carte de clavier à utiliser :

  Allemand (de-latin1-nodeadkeys)
  Amérique latine (la-latin1)
  Bélarusse (by)
  Belge (be2-latin1)
  Brésilien (br-abnt2)
  Brésilien (br-latin1)
  Britannique (uk)
  Bulgare (bg)
  Canadien français (cf)
  Croate (croat)
  Danois (dk-latin1)
  Dvorak
  Espagnol (es)
  Estonien (et)
  États-Unis (us)
  Finnois (fi-latin1)
  Français (fr-latin9)
  Grec (gr)
  Hébreu (hebrew)

    <Revenir en arrière>
```

FIGURE 7.4 – Choix du clavier

7.3 Détection du matériel

Après la sélection du pays et de la disposition du clavier, nous arrivons à la détection du matériel et au chargement des logiciels nécessaire à la suite de l'installation : cette étape est automatique et ne nécessite aucune action de votre part !

Chapitre 8

Configuration du réseau

Certaines cartes wifi nécessitent l'utilisation d'un micrologiciel propriétaire, qui ne peut donc pas être fourni avec l'installateur Debian. Si vous disposez d'une telle carte, l'installateur vous demandera si vous souhaitez charger ce *firmware* depuis un médium externe (clef USB...). L'installation par réseau wifi étant déconseillée, vous pouvez ignorer cette suggestion.

8.1 Si vous êtes connecté par Ethernet

L'installateur Debian essaie tout d'abord d'obtenir automatiquement une configuration réseau par DHCP. Si vous êtes connectés à une *Box ou à un réseau local configuré pour cela, vous n'aurez donc aucun réglage à faire à cette étape.

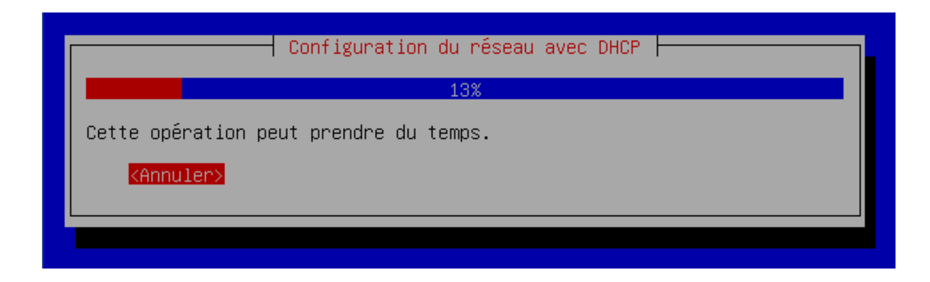


FIGURE 8.1 – Configuration du réseau par DHCP

S'il n'y a pas de serveur DHCP sur votre réseau local ou si la configuration par DHCP n'a pas marché, vous devrez configurer vous-même le réseau. L'installateur vous demandera alors successivement :

1. l'adresse IP de l'ordinateur,
2. le masque de sous-réseau,
3. l'adresse de la passerelle,
4. les adresses des serveurs DNS.

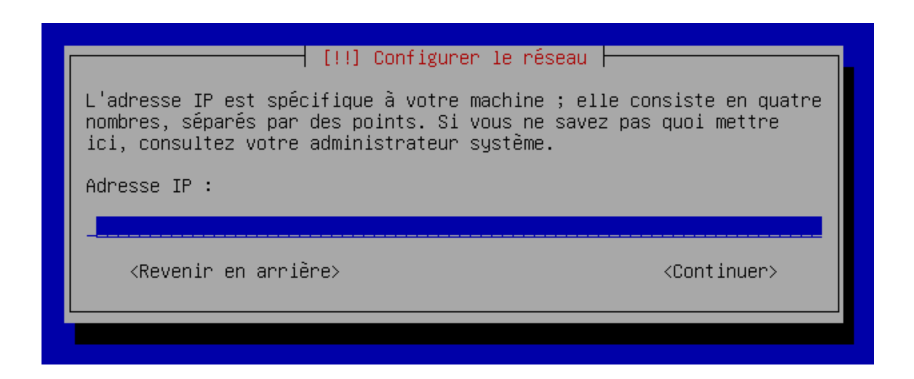


FIGURE 8.2 – Saisie de l'adresse IP

8.2 Si vous n'êtes pas connecté par Ethernet

Sélectionnez *Ne pas configurer le réseau maintenant*.

8.3 Le choix du nom de machine

Après la détection de votre matériel, une nouvelle boîte de dialogue apparaît : elle vous demande le nom de la machine et son domaine. Par défaut, le nom est debian. Libre à vous de changer pour un nom un peu plus en adéquation avec votre environnement.

Chapitre 9

Le réglage des comptes et mots de passe

9.1 Le mot de passe root

Qui est Monsieur Root ?

Linux est un système d'exploitation multi-utilisateurs. Chaque utilisateur a son login et son mot de passe personnel, et il existe un système de gestion des droits pour les fichiers et les répertoires. Un seul utilisateur a les pleins pouvoirs : c'est Monsieur Root, aussi appelé *super utilisateur* ou simplement *root*.

Entrer le mot de passe root

On vous demande de rentrer à deux reprises le mot de passe root. Il faut choisir un mot de passe complexe car celui qui le devine détient tous les droits sur la machine ! N'oubliez pas d'activer le verrouillage numérique si vous tapez des chiffres sur le pavé numérique.

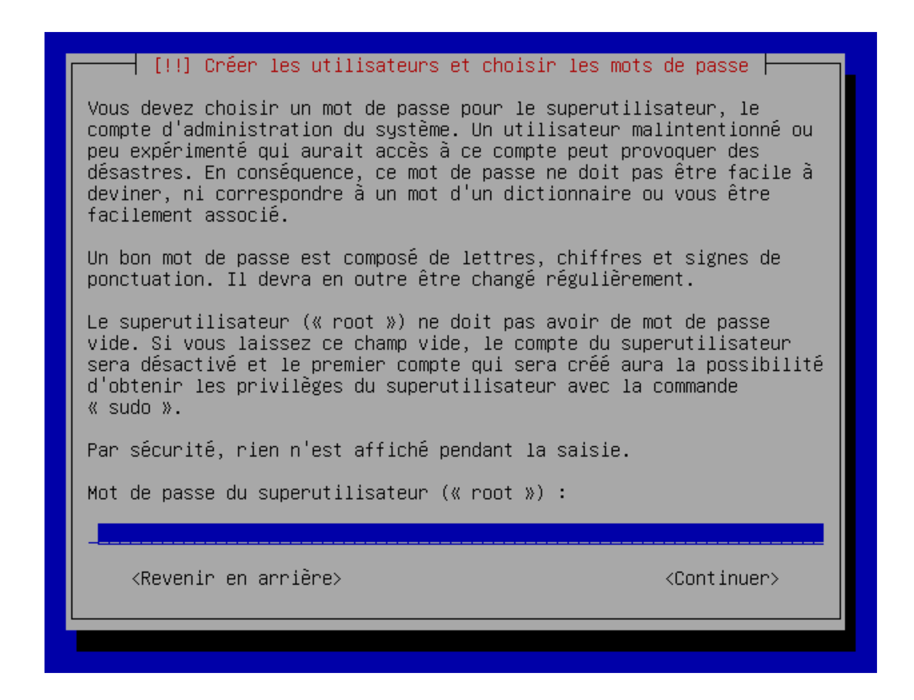


FIGURE 9.1 – Première demande du mot de passe root

9.2 Créer un compte utilisateur

Pourquoi ?

L'étape suivante est la création de votre propre compte utilisateur. En effet, l'utilisation du compte root est réservée à la modification de la configuration du système, à l'installation de paquets et aux rares tâches qui nécessitent les droits de root ; pour toutes les autres tâches, il faut utiliser un compte utilisateur.

Pourquoi ? Parce que l'utilisation du compte root est dangereuse : une fausse manipulation peut détruire le système... ce qui est impossible en tant que simple utilisateur !

Création du compte

Il vous demande d'abord le nom complet du compte (par exemple votre prénom et votre nom de famille), le login, puis d'entrer deux fois le mot de passe. Comme pour le root, utiliser un mot de passe non trivial est fortement conseillé.

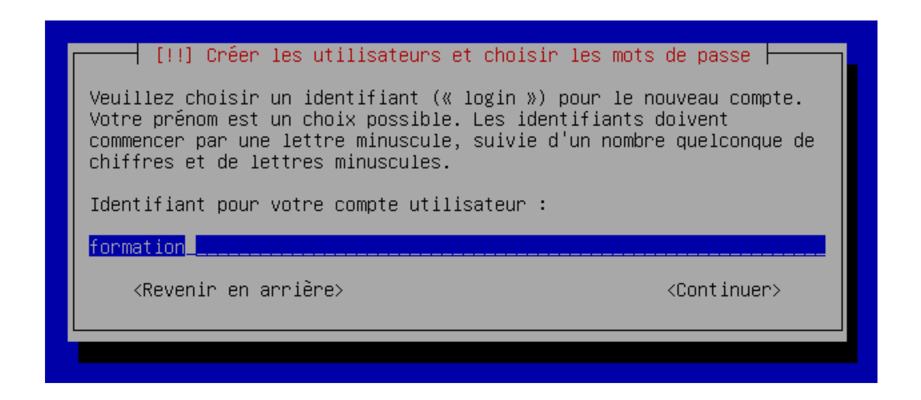


FIGURE 9.2 – Demande du login

9.3 Réglage de l'horloge

Après cela, l'installateur Debian interroge tout seul un serveur de temps pour régler de façon très précise votre horloge. Si votre connexion réseau ne fonctionne pas encore ou si un pare-feu bloque les requêtes d'heure, il vous faudra attendre un moment avant qu'il ne finisse par y renoncer.

Chapitre 10

Le système de fichiers

Avant d'aborder l'étape de partitionnement, il faut assimiler quelques notions sur la gestion des systèmes de fichiers sous Unix.

10.1 Périphérique, partition, système de fichiers

Périphériques et partitions

Un ordinateur travaille en utilisant des données présentes sur des périphériques : il peut s'agir des disques durs, de disquettes, de clefs USB ou encore de disques optiques.

Le cas des disques durs et des clefs USB est particulier. En effet, ces périphériques son rarement utilisés directement, mais sont plutôt découpés en *partitions*, qui apparaissent comme plusieurs périphériques plus petits. Dans le cas où un disque dur est utilisé comme un périphérique unique, il est en réalité « découpé » en une unique partition qui occupe la totalité de l'espace disponible, à l'exception d'une zone située au début du disque, qui constitue la *table de partitions*.

En définitive, les périphériques habituellement utilisés se présentent au système d'exploitation sous la forme d'une succession de blocs de données numérotés. Cette organisation basique ne définit aucune notion de fichier ou de répertoire, et permet seulement de stocker des données les unes à la suite des autres.

Système de fichiers

Les données sont normalement présentées à l'utilisateur et aux programmes selon une organisation structurée, sous la forme de répertoires et de fichiers. Pour pouvoir stocker ces données structurées sur un périphérique, il faut utiliser un format qui les représente sous la forme d'une succession de blocs de données : c'est ce qu'on appelle un *système de fichiers*.

Les systèmes de fichiers les plus courants sont la FAT (disquettes et clefs USB), NTFS (Windows), Ext2, Ext3 et Ext4 (Linux), ISO 9660 (CD) et UDF (DVD).

10.2 L'arborescence

Contrairement au système de fichiers Windows, il n'existe pas de lecteurs `A:`, `C:`, etc.

L'entrée du système de fichiers se situe à la racine, notée `/`.

Ensuite, il existe un certain nombre de répertoires présents par défaut. Le Tableau 10.1 explique les fonctions des plus importants d'entre eux (pour plus de détails, vous pourrez regarder le manuel **man hier** une fois votre installation effectuée).

Note

Pour certaines installations, il peut être intéressant de dédier un médium (partition de disque dur) à certains répertoires, en les séparant ainsi du médium principal de stockage de votre système. La troisième colonne identifie les répertoires pour lesquels une telle disposition *peut* présenter un intérêt : n'en déduisez pas que vous devez ainsi externaliser tous ces répertoires !

10.3 Les périphériques

L'accès aux périphériques

Une des originalités des systèmes Unix est leur manière d'accéder aux périphériques. Chaque périphérique du système (souris, disque dur, lecteur de CD, carte son, etc.) est représenté par un fichier spécial (Tableau 10.2). Écrire dans un tel fichier va envoyer des commandes au périphérique. Lire un tel fichier permet d'en recevoir des données. C'est une méthode très simple qui a fait ses preuves !

Les partitions

Pour connaître la position de vos disques durs IDE ou SATA et de vos lecteurs de CD (*primary master*, *primary slave*, *secondary master* ou *secondary slave*), le plus simple est de regarder dans le BIOS.

Vous pouvez aussi le savoir à partir des branchements des nappes IDE et des cavaliers sur les disques durs ou les lecteurs de CD : *primary* correspond à la première nappe IDE, et *secondary* à la seconde ; sur chaque nappe, on peut brancher au plus deux périphériques, un *master* et un *slave* (cela se règle avec un cavalier sur le périphérique).

Sur un disque dur IDE ou SATA, les partitions sont numérotées de la façon suivante :

Exemples :

- Si vous avez 4 partitions primaires, elles sont numérotées dans l'ordre `hda1/sda1` (hda1 pour un disque IDE / sda1 pour un disque SATA), `hda2/sda2`, `hda3/sda3` et `hda4/sda4`.
- Si vous avez dans l'ordre : 2 partitions primaires, 1 partition étendue avec 3 lecteurs logiques dedans, et 1 dernière partition primaire à la fin, ça donne :
  - Les deux premières partitions primaires sont `hda1/sda1` et `hda2/sda2`,
  - La partition étendue est `hda3/sda3`,
  - Les lecteurs logiques de la partition étendue sont, dans l'ordre, `hda5/sda5`, `hda6/sda6` et `hda7/sda7`,
  - La dernière partition primaire est `hda4/sda4`.

Les périphériques spéciaux

Il existe un certain nombre de périphériques « spéciaux » qui ne correspondent à aucun matériel, mais qui servent quand même !

Répertoire	description	intérêt d'un système de fichiers dédié
/	Répertoire "racine", point d'entrée du système de fichiers	oui (obligatoire)
/boot	Répertoire contenant le noyau Linux et l'amorceur	non (sauf cas très particuliers)
/bin	Répertoire contenant les exécutables de base, comme par exemple cp, mv, ls, etc.	non
/dev	Répertoire contenant des fichiers spéciaux nommés *devices* qui permettent le lien avec les périphériques de la machine	oui (automatiquement mis en place)
/etc	Répertoire contenant les fichiers de configuration du système	oui (délicat)
/home	Répertoire contenant les fichiers personnels des utilisateurs (un sous-répertoire par utilisateur)	oui
/lib	Répertoire contenant les librairies et les modules du noyau (/lib/modules)	non
/media	Répertoire contenant les « points de montage » des médias usuels : CD, DVD, disquette, clef USB	non
/root	Répertoire personnel de l'administrateur	non
/sbin	Répertoire contenant les exécutables destinés à l'administration du système	non
/tmp	Répertoire contenant des fichiers temporaires utilisés par certains programmes	oui
/usr	Répertoire contenant les exécutables des programmes (`/usr/bin` et `/usr/sbin`), la documentation (`/usr/doc`), et les programmes pour le serveur graphique (`/usr/X11R6`).	non (en général)
/var	Répertoire contenant les fichiers qui servent à la maintenance du système (les fichiers de journaux notamment dans `/var/log`)	oui

TABLE 10.1 – L'arborescence d'un système Linux

Fichier	**périphérique**
/dev/input/mouse0	souris
/dev/fd0	lecteur de disquettes
/dev/hda	lecteur maître de la première nappe IDE
/dev/hdb	lecteur asservi de la première nappe IDE
/dev/hdc	lecteur maître de la seconde nappe IDE
/dev/hdd	lecteur asservi de la seconde nappe IDE
/dev/sda	premier disque dur SATA, SCSI ou USB
/dev/sdb	second disque dur SATA, SCSI ou USB
/dev/ttyS0	premier port série ou infrarouge
/dev/ttyS1	second port série ou infrarouge
/dev/parport0	port parallèle
/dev/lp0	imprimante parallèle
/dev/usb/lp0	imprimante USB

TABLE 10.2 – Exemples de périphériques

Type	**ordre**	**numéros**
primaires	apparition sur le disque	de 1 à 4
lecteurs logiques	apparition dans la partition étendue	de 5 à 20

TABLE 10.3 – La numérotation des partitions

Fichier	**description**
/dev/null	on peut envoyer une infinité de données à ce périphérique, qui les ignorera...
/dev/zero	on peut lire une infinité de zéros depuis ce périphérique
/dev/random	on peut lire des nombres aléatoires depuis ce périphérique

TABLE 10.4 – Exemple de périphériques spéciaux

10.4 Intégration d'un système de fichiers (montage)

Considérons deux partitions (Figure 10.1). Sur ces partitions sont écrits deux *systèmes de fichiers* : ce sont des formats de stockage d'une arborescence de fichiers et de répertoires. La première partition contient une arborescence racine, et la seconde des répertoires personnels d'utilisateurs.

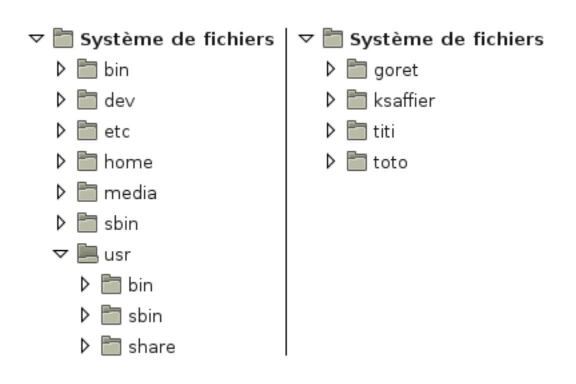


FIGURE 10.1 – Avant intégration

Nous allons pouvoir intégrer le second système de fichiers dans le répertoire `/home` du premier à l'aide de la commande **mount** : on parle de *monter* le second système de fichiers dans le *point de montage* `/home`.

Par exemple, si le deuxième système de fichiers est `/dev/hda2`, il suffira de taper :

```
# mount /dev/hda2 /home
```

pour obtenir la configuration présentée sur la Figure 10.2.



FIGURE 10.2 – Après intégration

Chapitre 11

Partitionner

11.1 Installation du système de base

Debian procède tout d'abord à l'installation du système de base : c'est l'ensemble des logiciels communs à toute installation de Debian. Cette étape ne nécessite aucune intervention de votre part et prend quelques minutes.

11.2 L'organisation des partitions sous Linux

Les types de systèmes de fichiers

Linux utilise deux types de systèmes de fichiers :

– *Swap* qui sert de *mémoire virtuelle*, qui est utilisée quand la mémoire vive est pleine ;
– *Ext4* qui sert à stocker les fichiers et les répertoires (il existe de nombreuses alternatives à Ext4 : Ext3, Ext2, ReiserFS, XFS, JFS...).

Découpage et dimensionnement

Traditionnellement, on crée une partition avec un système de fichiers de type *Swap* de taille :

– double ou triple de la taille de la mémoire vive quand celle-ci est inférieure à 256 Mo ;
– égale à la taille de la mémoire vive quand celle-ci est supérieure ou égale à 256 Mo.

Cette partition est appelée partition de swap ou d'échange.

Note

Les ordinateurs récents disposent d'une quantité proprement ahurissante de mémoire vive, souvent un giga-octet ou plus. C'est largement plus que vous n'en aurez besoin sous GNU/Linux, donc votre partition de *swap* aura peu de chance de servir ! Dans ce cas, vous pouvez quand même en définir une, puisque vous devriez également avoir de l'espace disque à revendre.

Pour stocker les fichiers et les répertoires, on crée souvent plusieurs partitions avec un système de fichiers de type *Ext4* (ou une de ses alternatives).

Pour les serveurs, les administrateurs GNU/Linux ont souvent pour habitude de sectionner le système de fichiers en de nombreuses partitions pour assurer une meilleure résistance du système aux pannes et aux failles. Par exemple, il ne faudrait pas qu'un simple utilisateur puisse saturer la partition sur laquelle se trouve la racine du système de fichiers juste en remplissant son répertoire personnel (`/home/son_login/`), car ceci pourrait rendre le système instable. Il ne faudrait pas non plus que les journaux système (ou logs) qui se trouvent dans le répertoire `/var/log/` remplissent la partition sur laquelle se trouve la racine suite à une attaque réseau, ce qui aurait la même conséquence. Ce raisonnement est valable pour plusieurs autres répertoires. Pour gagner en flexibilité, on utilise également le gestionnaire de volumes logiques (LVM), qui permet de définit des partitions redimensionnables à volonté.

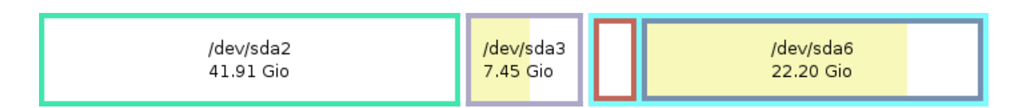


FIGURE 11.1 – Exemple de partitionnement

Sur une machine personnelle, de telles précautions ne sont pas nécessaires et imposent des contraintes inutiles sur la taille des répertoires. Il est en revanche utile de dédier une partition séparée aux fichiers personnels `/home/`, pour pouvoir réinstaller facilement votre système en conservant vos données, comme sur la Figure 11.1.

Deux exemples

Le Tableau 11.1 et le Tableau 11.2 donnent deux exemples de partitionnement : un pour une machine personnelle et un pour un serveur.

Partition	**taille**
Swap	double de la mémoire vive
/	7 Gio
/home	tout le reste de l'espace alloué à Linux

TABLE 11.1 – Pour un ordinateur personnel

Partition	**Taille**
Swap	Égale ou double de la mémoire vive
/	2 Gio
/tmp	500 Mio sur LVM
/var	1 Gio sur LVM
/home	selon les besoins des utilisateurs, sur LVM

TABLE 11.2 – Pour un serveur

Dans la suite de ce chapitre, nous allons mettre en œuvre la configuration proposée dans le premier exemple.

11.3 Repartitionner le disque dur

État initial

La procédure d'installation demande maintenant de choisir une méthode de partitionnement : répondez *manuel*. Il vous présente ensuite la table de partition actuelle de votre disque dur. La première ligne correspond au disque dur, et les lignes suivantes constituent la liste des partitions. Pour chaque partition, il est indiqué :

– le numéro de la partition ;
– le type de partition : *primaire* ou *logique* ;
– la taille,
– le système de fichiers : FAT 32, NTFS, Ext4, swap, etc.

Note

Parmi les partitions primaires, une partition peut être marquée comme « démarrable » : ce marqueur était autrefois utilisé pour déterminer la partition contenant le code à charger pour démarrer le système d'exploitation. Aujourd'hui, cette fonction est prise en charge par le chargeur de démarrage, dont nous parlerons plus tard : le marqueur de démarrage n'a donc plus qu'un rôle informatif, mais il est quand même présenté par l'outil de partitionnement.

Réduction de la partition Windows

Si un Windows est installé sur votre disque dur et que sa partition occupe tout le disque dur, alors il va falloir réduire la partition Windows pour libérer de l'espace à la fin du disque pour installer les partitions Linux.

Déplacez-vous vers le bas jusqu'à sélectionner la partition Windows puis appuyez sur **Entrée**. Vous accédez alors à un écran qui vous permet de changer les réglages de la partition. Sélectionnez *Taille* ; il vous demande la permission d'écrire les changements sur les disques avant de redimensionner les partitions ; répondez *Oui*. Ensuite, il vous indique quelle est la taille minimale possible (ce qui correspond à la taille occupée par les données existantes sur la partition Windows) et vous propose d'entrer la nouvelle taille que vous avez décidé d'allouer pour la partition Windows. Vous pouvez rentrer la taille en pourcentage de la taille maximale possible, mais je vous conseille plutôt de rentrer la vraie taille ; tapez par exemple **20.5 GB** si vous avez décidé d'allouer 20,5 Go ou **800 MB** si vous avez décidé d'allouer 800 Mo.

Création de la partition de Swap

Sélectionnez maintenant la ligne correspondant à l'espace libre et appuyez sur **Entrée**.

Choisissez ensuite l'action *Créer une nouvelle partition*. Il va alors vous demander :

1. la taille que vous avez choisie pour la partition de swap ;
2. le type de partition : Primaire ou Logique ;

3. l'emplacement de la partition : Début ou Fin (je vous conseille de choisir *Début* pour ne pas vous embrouiller).

Enfin, il vous affiche un écran qui récapitule les paramètres de la partition. Par défaut, il vous a probablement proposé de créer un *système de fichiers journalisé ext3* et / comme *Point de montage*. Sélectionnez la ligne *Utiliser comme : système de fichier journalisé ext3*, tapez **Entrée** puis sélectionnez *espace d'échange (« swap »)*. Vous revenez alors à l'écran récapitulatif des paramètres de la partition. Si tout vous semble bon, sélectionnez *Fin du paramétrage de cette partition* ; sinon, modifiez les paramètres qui ne correspondent pas à vos souhaits.

```
[!!] Partitionner les disques

Vous modifiez la partition n° 2 sur IDE1 maître (hda). Aucun système
de fichiers n'a été détecté sur cette partition.

Caractéristiques de la partition :

        Utiliser comme :         espace d'échange (« swap »)

        Indicateur d'amorçage :  absent

        Copier les données d'une autre partition
        Supprimer la partition
        Fin du paramétrage de cette partition

    <Revenir en arrière>
```

FIGURE 11.2 – Écran récapitulatif pour la partition de Swap

Note

La création d'une partition logique entraîne automatiquement la création de la partition étendue sous-jacente.

Création de la partition / (racine) au format Ext4

Sélectionnez de nouveau la ligne correspondant à l'espace libre et appuyez sur **Entrée**. Choisissez ensuite l'action *Créer une nouvelle partition*. Il va alors vous poser les mêmes questions que précédemment :

1. la taille que vous avez choisi pour la partition racine ;
2. le type de partition : Primaire ou Logique ;
3. l'emplacement de la partition : Début ou Fin.

Ensuite, il vous affiche l'écran qui récapitule les paramètres de la partition. Vérifiez :

– que le paramètre *Utiliser comme* est sur *système de fichiers journalisé ext4*,
– que le paramètre *Point de montage* est /,
– que les autres paramètres ont l'air bons.

puis sélectionnez *Fin du paramétrage de cette partition*.

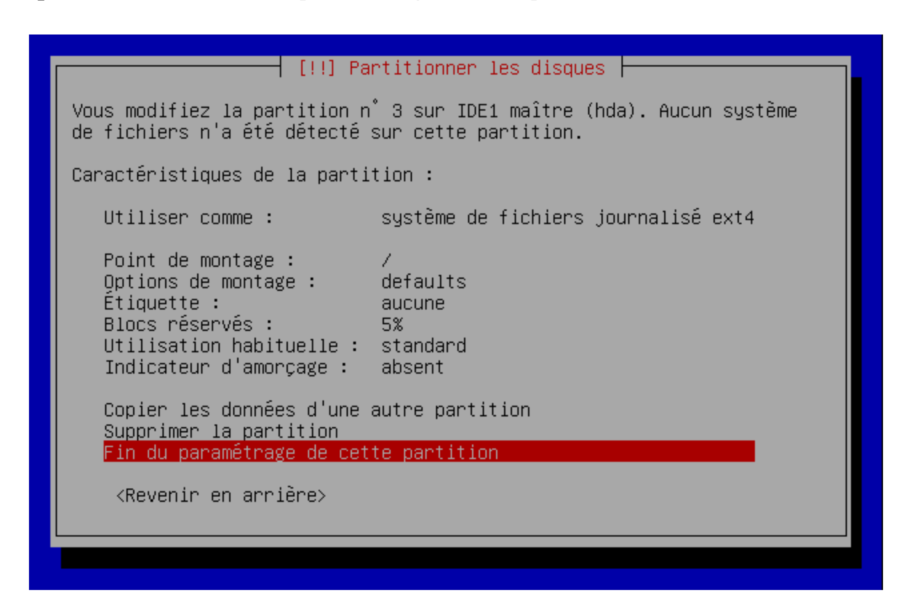


FIGURE 11.3 – Écran récapitulatif pour la partition root

Création de la partition /home au format Ext4

Sélectionnez de nouveau la ligne correspondant à l'espace libre et appuyez sur **Entrée**. Choisissez ensuite l'action *Créer une nouvelle partition*.

Procédez comme pour la partition racine, en donnant la taille souhaitée, et en vérifiant que le point de montage est bien `/home`, puis sélectionnez *Fin du paramétrage de cette partition*.

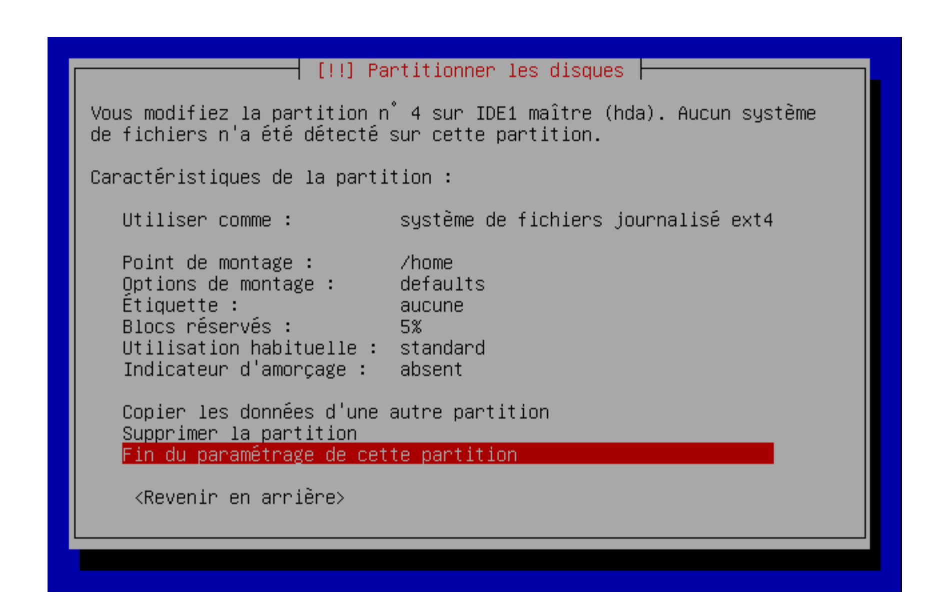


FIGURE 11.4 – Écran récapitulatif pour la partition /home

Appliquer la nouvelle table de partition et formater les nouvelles partitions

De retour à l'écran qui affiche la table des partitions, vérifiez que toutes les partitions sont à leur place, de la bonne taille et au bon format, puis sélectionnez *Terminer le partitionnement et appliquer les changements*. Un avertissement vous informera peut-être qu'aucun point de montage n'est affecté à la partition Windows, en vous demandant si vous souhaitez revenir au menu de partitionnement : répondez *Non*

```
[!!] Partitionner les disques

Voici la table des partitions et les points de montage actuellement
configurés. Vous pouvez choisir une partition et modifier ses
caractéristiques (système de fichiers, point de montage, etc.), un
espace libre pour créer une nouvelle partition ou un périphérique
pour créer sa table des partitions.

      Partitionnement assisté
      Configurer le RAID avec gestion logicielle
      Configurer le gestionnaire de volumes logiques (LVM)
      Configurer les volumes chiffrés

      IDE1 maître (hda) - 96.6 GB QEMU HARDDISK
           n° 1  primaire   40.0 GB    f  fat32
           n° 2  primaire  510.0 MB    f  swap     swap
           n° 3  primaire    7.0 GB    f  ext4     /
           n° 4  primaire   49.1 GB    f  ext4     /home

      Annuler les modifications des partitions
      Terminer le partitionnement et appliquer les changements

    <Revenir en arrière>
```

FIGURE 11.5 – Écran récapitulatif de la future table de partition

L'écran suivant est important. C'est après avoir validé cet écran que les changements vont être appliqués et que les nouvelles partitions seront formatées. Il faut donc bien lire les explications des actions qui vont être entreprises ; il est encore temps de revenir en arrière, après ce sera trop tard ! Si tout a l'air bon, répondez *Oui* à la question *Faut-il appliquer les changements sur le disque ?*.

Il crée alors la nouvelle organisation des partitions et formate la partition de Swap et la ou les nouvelle(s) partition(s) Ext4.

Chapitre 12

Les paquets

12.1 La source des paquets

Le système de gestion des paquets de Debian s'appelle APT (advanced package tool). Pour fonctionner, il peut avoir besoin de connaître la source des paquets susceptibles d'être installés. Il peut pour cela utiliser vos disques d'installation (CD ou DVD), mais il vous propose également d'utiliser un « miroir réseau », c'est à dire un dépôt de paquets accessible par une connexion réseau.

Si vous utilisez un CD ou un DVD d'installation complet, l'installateur vous demande si vous disposez d'un autre disque à analyser. Si c'est le cas, éjectez le premier disque, remplacez le par le second, puis validez, et ainsi de suite jusqu'au dernier disque. Vous arrivez ensuite à la configuration d'un miroir réseau.

Si vous n'avez pas encore de connexion à Internet fonctionnelle

À la question *Faut-il utiliser un miroir sur le réseau ?*, répondez *Non*.

Si vous avez déjà une connexion à Internet fonctionnelle

Note

Si vous utilisez la méthode *netinst*, l'installateur ne vous pose même pas la question, et vous demande directement quel miroir utiliser, puisque l'installation ne peut pas se faire sans miroir.

À la question *Faut-il utiliser un miroir sur le réseau ?*, répondez *Oui*. Répondez aux écrans suivants avec les consignes ci-dessous :

Sélection du miroir : sélectionnez votre pays ou un pays proche puis le nom d'un miroir dans la liste qu'il vous propose... sauf si vous avez connaissance d'un miroir Debian sur votre réseau local. Dans ce dernier cas, sélectionnez dans la liste : *Saisie manuelle*. Il vous demande alors le nom DNS du miroir Debian et le répertoire où se trouve le miroir (il vous propose par défaut le répertoire standard `/debian`).

Mandataire (proxy) HTTP :

- si vous n'avez pas de proxy pour accéder à Internet, laissez le champ vide et validez.
- si votre proxy ne requiert pas d'authentification par login et mot de passe, tapez :

```
http://proxy.exemple.org:8080
```

où *proxy.exemple.org* est le nom DNS de votre proxy et *8080* son port.

– si votre proxy requiert une authentification par login et mot de passe, tapez :

```
http://login:password@proxy.exemple.org:8080
```

où *proxy.exemple.org* est le nom DNS de votre proxy, *8080* son port, *login* et *password* votre login et mot de passe pour le proxy.

12.2 Popularity-contest

L'installateur charge les listes de paquets, puis vous demande si vous souhaitez participer aux statistiques d'utilisation des paquets. Répondez comme vous le souhaitez, cela n'influe en rien sur la configuration du reste du système.

12.3 Installation de logiciels supplémentaires

Pour créer un type de configuration, l'installateur de Debian propose des *tâches* pour des usages particuliers : *serveur web, serveur de courrier*. . . Cochez seulement *Utilitaires standards du système* (avec la touche **Espace**, et validez. Nous allons installer nous-mêmes tous les paquets dont nous avons besoin, d'une part pour apprendre, et d'autre part, parce que nous pouvons ainsi faire du sur mesure !

Note

Si vous êtes vraiment pressé et que vous souhaitez obtenir un système fonctionnel sans vous poser de question, vous pouvez sélectionner l'environnement de bureau, les utilitaires standards et éventuellement les outils destinés aux ordinateurs portables. Vous pourrez ainsi sauter de nombreux chapitres de cette formation, mais vous perdrez par la même occasion les connaissances que leur pratique apporte. . .



FIGURE 12.1 – Écran de sélection de logiciels

Ensuite, il procède à l'installation de nombreux paquets de base. Vous n'avez rien à faire pendant le déroulement de cette étape, qui prend quelques bonnes minutes.

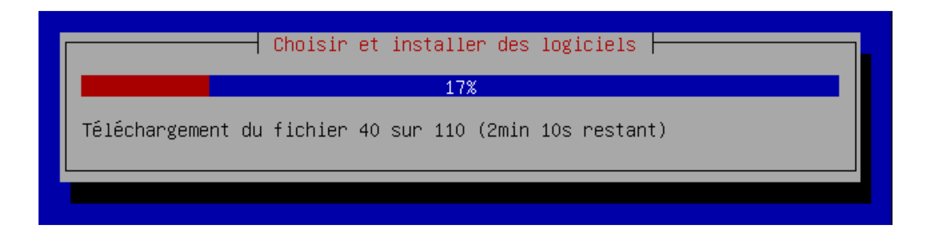


FIGURE 12.2 – Installation de paquets de base

Chapitre 13

Premier démarrage !

13.1 Installation du bootloader

Pour préparer le premier démarrage sous Linux, il faut rendre votre nouveau système d'exploitation démarrable directement depuis le disque dur. Pour cela, le programme GRUB va être installé dans le MBR (master boot record) de votre disque dur. C'est ce programme qui va vous proposer de choisir un des multiples systèmes d'exploitation installés sur votre ordinateur (et par la suite il vous permettra aussi de choisir la version du noyau Linux avec laquelle vous allez démarrer votre système Debian).

La procédure d'installation vous donne la liste des autres systèmes d'exploitation qui ont été détectés (si vous avez une partition Windows, vérifiez que Windows est bien mentionné dans la liste) et vous demande : *Installer le programme de démarrage GRUB sur le secteur d'amorçage ?*. Répondez *Oui*.

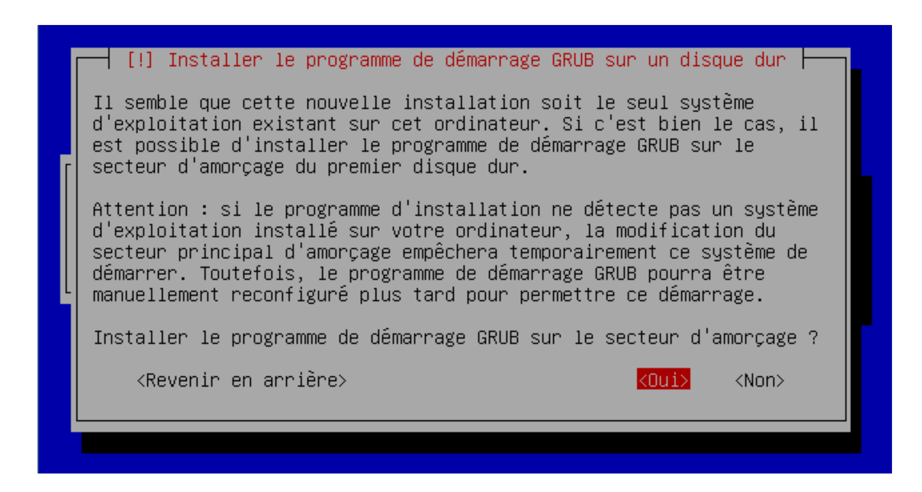


FIGURE 13.1 – Écran précédant l'installation de GRUB

13.2 Redémarrage et premier démarrage !

Le CD ou DVD va alors être éjecté automatiquement. Retirez-le du lecteur et faites *Continuer*.

[!!] Terminer l'installation
Installation terminée
L'installation est terminée et vous allez pouvoir maintenant démarrer votre nouveau système. Veuillez vérifier que le support d'installation (CD, disquettes) est bien retiré afin que le nouveau système puisse démarrer et éviter de relancer la procédure d'installation.
<Revenir en arrière> <Continuer>

FIGURE 13.2 – Écran précédant le redémarrage

L'ordinateur redémarre... Après le lancement du BIOS, GRUB se lance et vous affiche un menu bleu contenant la liste des systèmes d'exploitation qu'il peut démarrer. Si vous avez une partition Windows, cette dernière devrait apparaître dans la liste.

GNU GRUB version 0.97 (639K lower / 129984K upper memory)
Debian GNU/Linux, kernel 2.6.24-1-amd64
Debian GNU/Linux, kernel 2.6.24-1-amd64 (single-user mode)
Use the ↑ and ↓ keys to select which entry is highlighted
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.

FIGURE 13.3 – Écran de GRUB

13.3 Tester le multi-boot

Tester le démarrage de Windows

Si vous avez installé pendant la procédure d'installation un double boot Windows / Linux, c'est le moment ou jamais de tester si vous pouvez encore démarrer Windows !

Sélectionnez Windows avec les flèches Haut/Bas et appuyez sur **Entrée**.

Note

Certains ordinateurs sont préinstallés avec un logiciel de réinstallation du système d'exploitation imposé par le constructeur. Ce logiciel apparaît alors comme un *second Windows* dans le menu de GRUB. Vous pouvez alors le lancer depuis GRUB, ce qui peut éventuellement être utile. . .

Attention

Certains systèmes de restauration sont capable de modifier votre installation sans prévenir. Si vous possédez un ordinateur conçu par des inconscients comme Asus, ne lancez pas ce système, sous peine de ne plus pouvoir démarrer vos systèmes d'exploitation. Le menu de GRUB présente les systèmes d'exploitation étrangers dans l'ordre des partitions du disque dur, ce qui permet d'identifier le système de restauration.

En cas de problème...

GRUB ne marche pas et vous voulez au plus vite pouvoir démarrer de nouveau sous Windows ? Suivez la procédure suivante :

1. Démarrez sur le disque d'installation de Windows et choisissez *Réparer ou récupérer une installation de Windows*.
2. Sélectionnez votre installation de Windows dans la liste des choix proposés puis rentrez votre mot de passe administrateur.
3. Au prompt, tapez :

   ```
   C:\WINDOWS> fixmbr
   ```

   et confirmez que vous voulez réécrire sur le MBR.
4. Redémarrez en tapant :

   ```
   C:\WINDOWS> exit
   ```

Deuxième partie

Utilisation et configuration de base de Debian GNU/Linux

La première partie de cette formation vous a permis de suivre toute la procédure d'installation. Cette deuxième partie a pour but d'expliquer un certain nombre de choses qu'il est possible de faire *en console*, c'est-à-dire dans l'interface en mode texte présentée Figure 13.4 qui est présente par défaut. Vous allez apprendre à vous servir des commandes de base, d'un éditeur de texte et de l'outil de gestion de paquets de Debian. Ce n'est que dans la troisième partie que vous apprendrez à installer un serveur graphique et les applications graphiques les plus courantes.

```
Debian GNU/Linux 4.0 debian tty1

debian login: formation
Password:
Linux debian 2.6.17-2-686 #1 SMP Wed Sep 13 16:34:10 UTC 2006 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
formation@debian:~$ _
```

FIGURE 13.4 – Une console ouverte

Chapitre 14

Débuter en console

14.1 Notions de base

Se connecter

Une fois que la procédure d'installation est terminée, vous arrivez à l'invite de connexion :

```
Debian GNU/Linux 4.0 debian tty1

debian login: _
```

FIGURE 14.1 – Invite de connexion

Pour vous connecter, vous avez le choix entre :

– Vous connecter en tant que *root* : tapez **root**, appuyez sur **Entrée**, ensuite tapez le mot de passe root que vous avez défini pendant la procédure d'installation et appuyez sur **Entrée**. Vous voyez alors apparaître un certain nombre de messages et enfin le prompt du *root* :

```
debian:~# _
```

FIGURE 14.2 – Prompt du *root*

Quand vous êtes ainsi connecté en tant que *root*, vous avez tous les droits sur le système.

– Vous connecter en tant que simple utilisateur : tapez le nom d'utilisateur que vous avez défini pendant la procédure d'installation, appuyez sur **Entrée**, ensuite tapez le mot de passe associé à cet utilisateur et appuyez sur **Entrée**. Vous voyez alors apparaître un certain nombre de messages et enfin le prompt de l'utilisateur :

```
tanguy@debian:~$ _
```

FIGURE 14.3 – Prompt de l'utilisateur *tanguy* sur la machine *debian*

Quand vous êtes ainsi connecté en tant que simple utilisateur, vous n'avez que des droits limités sur le système.

Attention

Rappel : l'utilisation du compte *root* est réservée à la modification de la configuration du système, à l'installation de paquets et aux rares tâches qui nécessitent les droits de root. Pour toutes les autres tâches, il faut utiliser un compte utilisateur. En effet, l'utilisation du compte *root* est dangereuse : une fausse manipulation peut détruire le système, ce qui est impossible en tant que simple utilisateur !

Convention

Dans toute la suite de cette formation, nous adopterons la convention suivante :

– les commandes qui devront être exécutées en tant que *root* auront un prompt # :

```
# commande_à_exécuter
```

– les commandes qui devront être exécutées en tant que *simple utilisateur* auront un prompt % ou $:

```
% commande_à_exécuter
```

Passer d'une console à une autre

Vous n'avez peut-être pas encore remarqué, mais vous disposez de plusieurs consoles. Au démarrage, vous arrivez sur la première console, appelée **tty1** (teletypewriter). Vous pouvez passer à la deuxième console (appelée **tty2**) avec la combinaison de touches Alt-F2. Pour revenir à la première console, utilisez la combinaison de touches Alt-F1. Vous pouvez aussi utiliser Alt-Flèche gauche et Alt-Flèche droite (ou les touches fenêtre) pour passer d'une console voisine à l'autre. Par défaut, il y a 6 consoles.

Se déconnecter

Pour vous déconnecter, et retourner à l'invite de connexion, vous pouvez saisir la commande **logout** puis valider avec **Entrée**, ou bien utiliser la combinaison de touches Ctrl-d.

14.2 Les commandes Unix

Les bases

AVERTISSEMENT

Je présente certaines notions de façon volontairement inexacte, pour les rendre plus simple à appréhender. Après avoir pris connaissance avec l'interpréteur de commandes (le *shell*), je vous conseille de prendre le temps de lire un guide plus complet, comme celui d'Isabelle Vollant. Vous y trouverez probablement des différences avec ce que j'écris : dans ce cas, fiez-vous à ce guide plutôt qu'à mon texte !

Lorsque vous utilisez un shell, chaque ligne que vous tapez est une commande, qui invoque un logiciel ou une commande interne du shell. Ces commandes peuvent accepter des *arguments*, qui sont alors séparés par des espaces. Ainsi, pour invoquer la commande **cp** avec deux arguments, on tape :

```
$ cp /usr/share/doc/bash/copyright copyright2
```

Par ailleurs, certaines commandes acceptent également des *options*, qui sont en général données comme premiers arguments, et précédées par des tirets. Voici par exemple une commande **ln** avec une option **-s** et deux arguments :

```
$ ln -s /usr/share/doc/bash/copyright copyright2
```

ASTUCE

À partir de maintenant, je vous invite à essayer chaque commande au fur et à mesure que je vous les présenterai : j'ai fait en sorte qu'elles puissent s'enchaîner.

Le répertoire courant

Toutes vos commandes sont exécutées dans le *répertoire courant*. Ce répertoire est indiqué dans le prompt : dans l'exemple suivant, la commande **ls** va travailler sur le répertoire `/usr/share` :

```
formation@debian:/usr/share$ ls
```

Lorsque vous commencez à utiliser un shell, après vous être connecté, le répertoire courant est fixé à votre répertoire personnel, `/home/<utilisateur>`, qui est abrégé par le symbole ~. Vous pouvez changer de répertoire avec la commande **cd** (*change directory*). Avec un argument, elle vous envoie dans le répertoire donné ; sans argument, elle vous ramène à votre répertoire personnel :

```
formation@debian:~$ cd /usr/share
formation@debian:/usr/share$ cd
formation@debian:~$
```

Utiliser les répertoires

Lorsque vous souhaitez agir sur un fichier, vous devez l'indiquer à la commande que vous utilisez pour cela. Il y a trois façons de désigner un fichier :

de façon absolue en indiquant son chemin complet, en commençant par la racine `/`, par exemple : `/usr/share/doc/bash/copyright` ;

de façon relative en indiquant son chemin par rapport au répertoire courant, *sans* commencer par une `/`, par exemple `doc/bash/copyright`, si je me situe dans le répertoire `/usr/share/` ;

par rapport à votre répertoire personnel en indiquant son chemin à partir de votre répertoire personnel, abrégé par le caractère ~, par exemple : `~/.bash_history`

Par ailleurs, vous pouvez *remonter d'un niveau* dans un chemin, en utilisant le répertoire spécial `..`, que l'on appelle *répertoire parent*. Ainsi, quand vous êtes dans votre répertoire personnel, vous pouvez passer dans le répertoire `/home/`, puis dans `/`, puis retourner dans votre répertoire personnel ainsi :

```
~$  cd ..
/home$  cd ..
/$ cd
~$
```

Enfin, vous pouvez désigner le répertoire courant en utilisant le répertoire spécial `.` :

```
~$  cd .
~$  (on change... vers le répertoire courant)
```

Note

Lorsque vous écrivez le chemin d'un fichier, les sous-répertoire composant ce chemin doivent être séparés par des barres obliques `/` et non des contre-obliques comme sous Windows et DOS. Un répertoire est un fichier, d'un genre particulier, et peut être désigné avec ou sans barre oblique finale : `/usr/share` est équivalent à `/usr/share/` (voire même à `/usr//share///` si ça vous amuse).

Vous pouvez créer un nouveau répertoire en utilisant la commande **mkdir** (*make directory*), et en précisant le chemin (absolu, relatif, ou relatif à votre répertoire personnel) du répertoire que vous voulez créer. Ainsi, ces quatre commandes sont équivalentes (la dernière est un exemple d'utilisation d'un répertoire parent) :

```
~$  mkdir test
~$  mkdir /home/formation/test
~$  mkdir ~/test
~$  mkdir ../formation/test
```

Enfin, vous pouvez effacer un répertoire avec la commande **rmdir** (*remove directory*) :

```
~$  mkdir test2
~$  rmdir test2
```

Utiliser des fichiers

Voici quelques commandes indispensables, à essayer et à apprendre :

cp copie un fichier vers un répertoire, en changeant éventuellement son nom :

```
$ cp /usr/share/doc/bash/copyright .
$ cp /usr/share/doc/bash/copyright copyright2
```

mv (*move*) déplace un fichier vers un répertoire, en changeant éventuellement son nom :

```
$ mv copyright test/
$ mv copyright2 test/copyright3
```

rm (*remove*) supprime un fichier :

```
$ rm test/copyright3
```

ls affiche le nom d'un fichier, le contenu d'un répertoire, ou du répertoire courant :

```
$ ls
test
$ ls test
copyright
$ ls test/copyright
test/copyright
```

cat (con*cat*éner) affiche le contenu d'un ou plusieurs fichiers (les uns à la suite des autres, sans séparation, dans ce cas) :

```
$ cat test/copyright
[la licence de Bash]
$ cat test/copyright test/copyright
[la licence de Bash en deux exemplaires]
```

more et less affiche le contenu d'un long fichier, avec la possibilité de naviguer dedans (avec la touche **Espace**, les flèches, et la touche **q** pour quitter).

Nettoyage

Nous avons déjà commencé à peupler votre répertoire personnel de fichiers sans intérêt… Il est temps de faire le ménage. Pour cela, nous allons utiliser une option spéciale de la commande **rm** : **-rf**, un raccourci pour **-r -f**, qui permet d'effacer un répertoire :
- en descendant dans le répertoire et tous ses sous-répertoires éventuels pour les effacer avant,
- en forçant l'effacement sans demander de confirmation à chaque fois.

Dans notre cas, nous avons un répertoire `test/` à effacer :

```
$ rm -rf test
```

Commandes incontournables

Si vous ne deviez retenir que deux commandes : **apropos** et **man**.

apropos

Cette commande permet... de chercher une commande, à partir d'un mot-clef ! Ainsi, si vous cherchez comment naviguer sur le Web :

```
% apropos web
w3m (1)              - a text based Web browser and pager
```

man

Cette commande permet d'afficher le manuel de n'importe quel commande, fichier de configuration, fonction C… installé sur votre système. Pour afficher le manuel de w3m que nous avons découvert grâce à la commande **apropos** :

```
% man w3m
```

Autres commandes utiles

Voici quelques commandes qui vous seront certainement utiles un jour ou l'autre : **ln**, **find**, **grep**, **chmod**, **chown** et **chgrp**.

Elles ne vous seront pas utiles tout de suite : pour les découvrir, je vous renvoie à leurs manuels respectifs (**man ln**, **man find**…) et au site d'Isabelle Vollant, qui les décrit de façon très pratique.

Les principales commandes système

mount

Une première explication de l'utilisation de cette commande a déjà été donnée dans la section 10.4 de la première partie de cette formation. Elle sert à intégrer un système de fichiers dans un autre. Elle doit être exécutée en tant que root. Sa syntaxe habituelle est :

```
# mount -o options /dev/periphérique /mnt
```

à condition que le type de système de fichiers soit supporté par le noyau et que le répertoire `/mnt` existe déjà.

Pour démonter ce système de fichiers, il suffit de taper en root :

```
# umount /mnt
```

Par contre, n'importe quel utilisateur peut taper la commande **mount** tout court pour savoir quels sont les systèmes de fichiers montés à l'instant d'exécution de la commande.

su

Cette commande sert à changer d'utilisateur, après avoir rentré le bon mot de passe, bien sûr !

su - permet de devenir root.

su - toto permet de devenir l'utilisateur *toto*.

Note
Le passage de root à un simple utilisateur par la commande **su toto** se fait sans rentrer le mot de passe de l'utilisateur *toto*.

ps

Cette commande sert à lister les processus et leurs propriétés. Sous Unix, chaque tâche s'exécute au sein d'un ou plusieurs processus. Chaque processus a un PID (Processus ID) qui lui est propre. Si un processus plante, les autres processus ne sont pas affectés. On peut tuer un processus avec la commande **kill** ou **killall**.

ps liste les processus de l'utilisateur qui exécute la commande qui sont rattachés au terminal depuis lequel la commande est exécutée.

ps -u liste les processus de l'utilisateur qui exécute la commande quel que soit le terminal de rattachement.

ps -au liste les processus de tous les utilisateurs quel que soit le terminal de rattachement.

ps -aux liste les processus de tous les utilisateurs même ceux qui ne sont rattachés à aucun terminal. Cette commande liste donc l'intégralité des processus du système. Elle est équivalente à la commande **ps -A**

ps -faux liste tous les processus du système en les regroupant par enchaînement d'exécution.

kill et pkill

Les commandes **kill** et **killall** servent à envoyer des signaux à des processus.

kill 42 envoie le signal *TERM* au processus dont le PID est 42. En gros, on demande au processus 42 de se terminer tout seul. Bien sur, on ne peut terminer que les processus que l'on a soi-même lancé, sauf le root qui peut faire ce qu'il veut avec tous les processus.

kill -9 42 envoie le signal *KILL* au processus dont le PID est 42. Quand un processus est planté, c'est le seul moyen de l'arrêter, car la commande précédente n'aura pas d'effet.

pkill vlc envoie le signal *TERM* au processus dont le nom est *vlc*. Cette commande est à répéter plusieurs fois s'il y a plusieurs processus qui portent le nom *vlc*.

pkill -9 vlc envoie le signal *KILL* au processus dont le nom est *vlc*.

Autres commandes système

passwd change le mot de passe (il commence par demander l'ancien mot de passe quand il s'agit d'un simple utilisateur).

groups pour savoir à quels groupes appartient l'utilisateur.

adduser toto ajoute l'utilisateur *toto* au système.

deluser toto supprime l'utilisateur *toto* du système.

adduser toto disk ajoute l'utilisateur *toto* au groupe *disk* (modification effective après que l'utilisateur *toto* se soit déconnecté puis reconnecté).

deluser toto audio enlève l'utilisateur *toto* du groupe *audio*.

df -h fait le point sur l'espace libre de chaque système de fichiers monté.

du -sh mesure la taille du répertoire depuis lequel il est exécuté.

halt éteint l'ordinateur.

reboot redémarre l'ordinateur.

uptime indique depuis combien de temps le système n'a pas redémarré. Certains s'amusent ainsi à faire des *concours d'uptime* pour prouver la stabilité de leur machine sous GNU/Linux !

w et who permet de savoir quels utilisateurs sont connectés sur le système et ce qu'ils font.

lspci, lsusb et lshw donnent des informations sur les périphériques connectés à votre système (PCI, AGP, USB ou autres) : très pratique pour avoir des renseignements sur son matériel ! Attention, quand la commande affiche *Unknown device*, cela veut juste dire que l'identifiant du périphérique n'a pas de nom correspondant dans la base de données de ces commandes, mais cela ne veut pas dire que le périphérique marche, ne marche pas ou ne marchera jamais sous Linux !

cat /proc/cpuinfo donne plein d'informations sur le processeur.

uname -a donne des informations sur le système, notamment la version du noyau.

Les petites commandes pratiques

date donne l'heure, selon l'horloge de votre ordinateur ;

cal affiche un calendrier du mois courant. **cal 2005** affiche un calendrier de l'année 2005 ;

bc une calculatrice en mode texte ;

Ctrl-l permet de rafraîchir l'affichage d'une application en console quand l'affichage est perturbé (par un message d'erreur par exemple) ;

Ctrl-s et Ctrl-q permettent respectivement de bloquer et de débloquer l'affichage d'un terminal.

Ctrl-c arrête un programme, ou annule une commande en cours de saisie

Chapitre 15

Récupérer les fichiers de configuration

Avant d'aller plus loin, nous allons récupérer les fichiers de configuration que je vous avais demandé de stocker sur votre partition Windows, sur un CD ou sur une disquette au Chapitre 5.

15.1 Copie depuis un média amovible

CD

Insérez le disque contenant les fichiers de configuration et montez-le :

```
# mount /media/cdrom0
```

Copiez l'archive contenant les fichiers de configuration :

```
# cp /media/cdrom0/fichiers-config.tar.gz ~
```

Une fois les fichiés copiés, démontez le disque :

```
# umount /media/cdrom0
```

Mettez-vous dans le home du root, puis décompressez l'archive :

```
# cd
# tar xvzf fichiers-config.tar.gz
```

Si vous le souhaitez, vous pouvez désormais supprimer l'archive contenant les fichiers de configuration :

```
# rm fichiers-config.tar.gz
```

Clef USB

Branchez votre clef USB. Attendez quelques secondes. Des messages vont apparaître à l'écran, en particulier :

```
sda: sda1
```

Notez le nom de la partition trouvée sur votre clef (ici, *sda1*), et montez votre clef :

```
# mkdir /media/clef0
# mount /dev/sda1 /media/clef0
```

Procédez ensuite comme pour un CD, en remplaçant *cdrom0* par *clef0*.

15.2 Copie à partir d'une partition Windows

Créez un répertoire destiné à accueillir la partition Windows :

```
# mkdir /media/win
```

Montez la partition Windows dans ce répertoire :

```
# mount /dev/partition /media/win
```

où `/dev/partition` désigne votre partition Windows (la désignation des partitions était expliquée dans la section 10.3 dans la première partie).

Copiez l'archive contenant les fichiers de configuration :

```
# cp /media/win/ ↩
   chemin_vers_le_répertoire_où_vous_aviez_placé_les/ ↩
   fichiers-config.tar.gz ~
```

Mettez-vous dans le home du root, puis décompressez l'archive :

```
# cd
# tar xvzf fichiers-config.tar.gz
```

Vous pouvez désormais si vous le souhaitez supprimer l'archive contenant les fichiers de configuration :

```
# rm fichiers-config.tar.gz
```

15.3 Copie à partir d'Internet

Si vous avez déjà une connexion Internet fonctionnelle, vous pouvez récupérer l'archive contenant les fichiers de configuration directement par HTTP. Téléchargez le fichier `fichiers-config.tar.gz` dans le home du root :

1. Si vous devez passer par un proxy pour accéder à Internet :
   – si votre proxy ne requiert pas d'authentification par login et mot de passe :

   ```
   # export http_proxy="http://proxy.exemple.org:8080"
   ```

   où *proxy.exemple.org* est le nom DNS de votre proxy et *8080* son port.
   – si votre proxy requiert une authentification par login et mot de passe :

   ```
   # export http_proxy="http://login:password@proxy. ↩
      exemple.org:8080"
   ```

   où *proxy.exemple.org* est le nom DNS de votre proxy, *8080* son port, *login* et *password* votre login et mot de passe pour le proxy.

2. Téléchargez le fichier et déplacez-le dans le home du root :

```
% wget http://formation-debian.via.ecp.fr/fichiers- ↩
    config.tar.gz
# mv fichiers-config.tar.gz ~
```

3. Mettez-vous dans le home du root, puis décompressez l'archive :

```
# cd
# tar xvzf fichiers-config.tar.gz
```

4. Vous pouvez désormais si vous le souhaitez supprimer l'archive contenant les fichiers de configuration :

```
# rm fichiers-config.tar.gz
```

Chapitre 16

Vim : un éditeur de texte

16.1 Un outil de base sous Linux

L'éditeur de texte est un outil de base sous Linux. Il sert notamment à modifier les fichiers de configuration du système. Les deux éditeurs de texte les plus connus et les plus utilisés sont Vim et Emacs. Et comme je ne connais pas *Emacs*, eh bien je vais vous expliquer comment fonctionne Vim !

VIM signifie *Vi IMproved* : il s'agit d'une version améliorée du classique vi. Il est très complet, peu gourmand en ressources, et fait très bien la coloration syntaxique. Il n'est pas facile à maîtriser au début, mais vous serez rapidement conquis !

16.2 Installer et configurer Vim

Installation de Vim

Il va falloir installer les paquets permettant de faire marcher Vim. Pour l'instant, vous ne savez pas encore installer des paquets et je vous propose donc de suivre les instructions suivantes sans trop comprendre.

```
# aptitude install vim
```

Installation du fichier de configuration

Remplacez le fichier de configuration installé par défaut par mon fichier de configuration :

```
# cp ~/fichiers-config/vimrc /etc/vim/
```

ou, si vous ne suivez pas ma formation depuis le début :

```
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
    vimrc
# mv vimrc /etc/vim/
```

Ce fichier deviendra le fichier de configuration de Vim par défaut pour tous les utilisateurs. Un utilisateur pourra aussi utiliser son propre fichier de configuration en le mettant dans son répertoire personnel sous le nom `.vimrc`.

16.3 Se servir de vim

Pour éditer un fichier texte existant ou créer un nouveau fichier texte, il suffit de taper :

```
# vim nom_du_fichier
```

```
utilisateur pourra aussi utiliser son propre fichier de configuration en
le mettant dans son home (même nom mais précédé d'un point).</para>

</simplesect>
</sect1>

<sect1 id="vim-util"><title id="tvim-util">Se servir de vim</title>

<para>Pour éditer un fichier texte existant ou créer un nouveau fichier
texte, il suffit de taper : </para>

<screen>
<prompt># </prompt><userinput>vim nom_du_fichier</userinput>
</screen>

<figure><title>Vim</title>
<mediaobject>
  <imageobject>
    <imagedata fileref="fichiers/vim.png"/>
  </imageobject>
</mediaobject>
</figure>
vim.xml                                                        123,41 46%
"vim.xml" [converti] 245L, 10358C écrit(s)
```

FIGURE 16.1 – Vim

Tout d'abord, il faut comprendre qu'il existe plusieurs modes de fonctionnement :

– Le mode *Commande*, dans lequel vous vous trouvez quand vous ouvrez Vim. Dans ce mode, vous tapez des commandes... que nous verrons plus loin ! Si vous êtes dans un autre mode et que vous voulez revenir au mode commande, tapez **Echap**.
– Le mode *Insertion* auquel on accède par la touche **Inser**. L'indicateur `-- INSERTION --` apparaît alors en bas de l'écran. Dans ce mode, vous insérez du texte classiquement.
– Le mode *Remplacement* auquel on accède en appuyant une deuxième fois sur **Inser**. L'indicateur `-- REMPLACEMENT --` apparaît alors en bas de l'écran. Dans ce mode, le texte entré remplace le texte présent sous le curseur.
– Le mode *Visuel* auquel on accède par la touche **v** depuis le mode Commande. L'indicateur `-- VISUEL --` apparaît alors en bas de l'écran. Ce mode permet de sélectionner du texte pour y appliquer globalement des commandes.

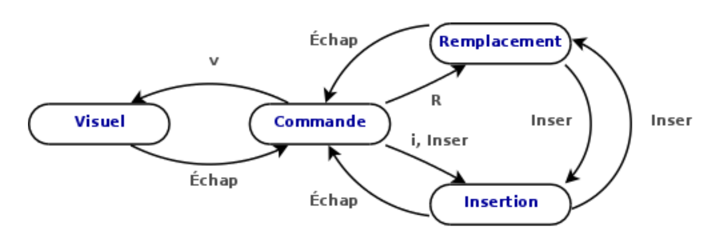


FIGURE 16.2 – Comment passer d'un mode à un autre ?

Voici une liste des commandes les plus utilisées. Il faut bien entendu être en mode **Commande** pour les taper :

- **:h** pour accéder à l'aide ;
- **:w** pour enregistrer ;
- **:w nom_du_fichier** pour faire *enregistrer-sous* `nom_du_fichier` ;
- **:q** pour quitter ;
- **:wq** ou **:x** pour enregistrer et quitter ;
- **:q!** pour quitter sans enregistrer les modifications ;
- **:r** pour inclure le contenu d'un autre fichier ;
- **/mot_clef** pour rechercher un mot vers le bas du texte (**n** pour passer à l'occurrence suivante, **N** pour passer à la précédente) ;
- **?mot_clef** pour rechercher un mot vers le haut du texte (**n** pour passer à l'occurrence suivante, **N** pour passer à la précédente (donc vers le bas)) ;
- **:numéro_de_ligne** pour aller directement à cette ligne ;
- **nombre_de_lignes yy** (sans espace) pour copier ce nombre de ligne à partir du curseur (**yy** pour copier une ligne et **y** pour copier un groupe de mots en mode visuel) ;
- **nombre_de_lignes dd** (sans espace) pour couper ce nombre de ligne à partir du curseur (**dd** pour couper une ligne en mode commande et **d** pour couper un groupe de mots en mode visuel) ;
- **p** pour coller après le curseur, **P** pour coller avant le curseur ;
- **u** pour annuler la dernière modification. Vous pouvez appuyer plusieurs fois sur **u** pour annuler *les* dernières modifications ;
- Ctrl-r ou **:redo** pour annuler la dernière annulation. Vous pouvez renouveler la combinaison de touches pour annuler les annulations antérieures ;
- **:%s/toto/tata/g** pour remplacer toutes les occurrences de la chaîne de caractères *toto* par la chaîne de caractère *tata*.

Pour plus d'informations sur Vim, je vous invite à consulter le wiki francophone sur Vim.

16.4 Editer un fichier de configuration Unix

Sous Unix, et en particulier sous Linux, la configuration du système et des programmes se fait très souvent en éditant des fichiers textes qui contiennent des paramètres de configuration. Ces paramètres de configuration suivent une certaine syntaxe, différente pour chaque programme, et que l'utilisateur doit connaître. Généralement, il y a une instruction de configuration par ligne de texte. Le système ou le programme va alors lire son ou ses fichier(s) de configuration et s'adapter à la configuration demandée.

Presque tous les programmes et systèmes Unix sont conçus avec une règle qui dit qu'il ne tient pas compte des lignes du fichier de configuration qui commencent par un certain caractère (souvent **#**). L'utilisateur peut alors mettre des lignes de commentaires dans le fichier de configuration en commençant ces lignes par le caractère particulier. Il peut aussi facilement activer ou désactiver une ligne du fichier de configuration en enlevant ou en ajoutant le caractère particulier au début de la ligne. Le fait de désactiver ainsi une ligne de configuration se dit « *commenter une ligne* » et le fait d'activer ainsi une ligne de configuration se dit « *décommenter une ligne* ». Ces expressions seront régulièrement utilisées dans la suite de cette formation.

Note

Quand vous éditez un fichier de configuration existant, il est généralement très facile de savoir quel est le caractère particulier : les lignes de commentaires sont nombreuses, contiennent souvent des phrases rédigées et apparaissent normalement en rouge sous Vim.

Chapitre 17

Faire marcher la connexion à Internet

L'installation des pilotes du modem et la configuration de la connexion dépendent du modèle de votre modem. Avec un peu de chance, vous trouverez une section ci-dessous spécifique à votre modem.

Note

Si vous avez une connexion ADSL avec un modem Ethernet ou USB où la configuration se fait par DHCP (c'est le cas si vous êtes branché à une FreeBox en Ethernet par exemple), votre connexion Internet est déjà configurée : vous pouvez passer directement au chapitre suivant.

17.1 Connexion par modem ADSL Ethernet ou modem câble Ethernet en PPPoE

Il vous faut tout d'abord installer *pppoeconf*, le logiciel qui va nous permettre de configurer votre connexion :

```
# aptitude install pppoeconf
```

Ensuite, il faut faire marcher la liaison vers votre fournisseur d'accès, qui est de type PPPoE (point to point protocol over Ethernet). Pour cela, lancez l'assistant et répondez à ses questions :

```
# pppoeconf
```

Répondez aux questions en lisant les messages avec attention :

1. *Tous les périphériques ont-ils été trouvés ?* Si vous avez une seule carte réseau, et si son module est bien chargé, il doit afficher *J'ai trouvé 1 périphérique ethernet : eth0.* Répondez *Oui*.
2. Il part ensuite à la recherche d'un concentrateur PPPoE... et si tout va bien, il annonce *J'ai trouvé un concentrateur d'accès sur eth0. Dois-je configurer PPPoE pour cette connexion ?* Répondez *Oui*.

3. Ensuite, il vous met en garde contre un écrasement du fichier de configuration `/etc/ppp/peers/dsl-provider` : répondez *Oui*, même si vous n'avez pas de copie de sauvegarde !
4. S'ensuit une question au sujet des options *noauth* et *defaultroute* : répondez *Oui*.
5. *Entrez le nom d'utilisateur* : tapez le login qui vous a été attribué par votre fournisseur d'accès (*login@fournisseuradsl*, en fait).
6. *Entrez le mot de passe* : tapez le mot de passe associé.
7. *Utilisation du serveur de nom associé ?* Suivez le choix recommandé : répondez *Oui*.
8. *Problème de* MSS *restreint* : si vous n'êtes pas un expert réseau, vous ne comprendrez probablement pas grand chose à cette question : suivez-donc encore une fois le choix recommandé, i.e. répondez *Oui*.
9. *Voulez-vous que la connexion soit établie au démarrage de la machine ?* Répondez selon votre utilisation habituelle de la connexion Internet.
10. *Voulez-vous démarrer la connexion tout de suite ?* C'est l'occasion de tester : répondez *Oui* !

Si vous avez mal répondu à une des questions, relancez l'assistant :

```
# pppoeconf
```

Comme expliqué au dernier écran, pour établir la connexion (si elle n'est pas lancée au démarrage), lancez :

```
# pon dsl-provider
```

et pour la terminer, tapez :

```
# poff
```

17.2 Connexion par modem classique

AVERTISSEMENT
Je ne dispose pas de modem classique, les explications qui suivent sont donc peut-être obsolètes.

Cette section explique comment se connecter à Internet avec un modem classique branché sur une ligne téléphonique classique. La procédure ci-dessous doit marcher sans problème avec un modem externe branché sur port série, ou avec un modem PCMCIA ; par contre, pour les modems PCI ou les modems intégrés, la procédure est différente et dépend de chaque modem... et n'est pas expliquée dans ce document.

Il vous faut tout d'abord installer *pppconfig*, le logiciel qui va nous permettre de configurer votre connexion :

```
# aptitude install pppconfig
```

Si c'est un modem PCMCIA...

Installer le paquet *pcmciautils* :

```
# aptitude install pcmciautils
```

Si c'est un modem externe sur port série...

Regardez sur quel port série le modem est branché :
- s'il est connecté sur le port série COM1, le device correspondant sera `/dev/ttyS0` ;
- s'il est connecté sur le port série COM2, le device correspondant sera `/dev/ttyS1`.

Vérifier que le port série marche

Si c'est un modem PCMCIA, insérez le carte dans votre portable ; si c'est un modem externe, allumez-le. Vous allez maintenant vérifier que le système a bien reconnu le port série :

```
# setserial /dev/ttyS0
/dev/ttyS0, UART: 16550A, Port: 0x03e8, IRQ: 0
```

- Si la ligne qui s'affiche contient **UART: 16550A**, alors cela signifie que le port série est bien reconnu.
- Si, par contre, la ligne qui s'affiche contient **UART: unknown**, alors cela signifie que le port série n'est pas reconnu (et là je ne sais pas trop ce qu'on peut faire...).

Configurer la connexion vers le fournisseur d'accès

Le plus simple pour configurer la connexion vers votre fournisseur d'accès est d'utiliser l'assistant qui est installé par défaut :

```
# pppconfig
```

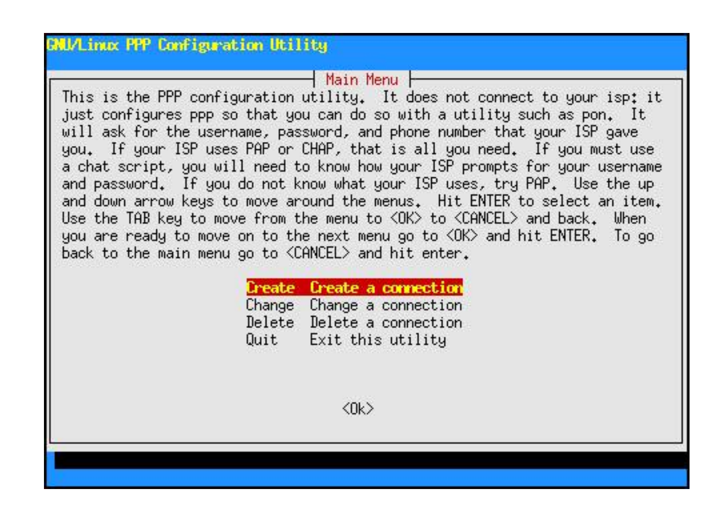


FIGURE 17.1 – Premier écran de pppconfig

Sélectionnez *Create - Create a connection* et répondez aux questions successives :

1. *Provider Name* : rentrez un nom pour cette connexion (par exemple le nom de votre fournisseur d'accès Internet) ;
2. *Configure Nameservers (DNS)* : sélectionnez *Use dynamic DNS* pour obtenir automatiquement les adresses des serveurs DNS de votre fournisseur d'accès à chaque connexion ;
3. *Authentication Method* : sélectionnez *PAP Peer Authentication Protocol* ;
4. *User Name* : tapez le login qui vous a été attribué par votre fournisseur d'accès (tapez-le entre guillemets si le login contient des caractères de ponctuation) ;
5. *Password* : tapez le mot de passe qui vous a été donné par votre fournisseur d'accès (tapez-le entre guillemets si le mot de passe contient des caractères de ponctuation) ;
6. *Speed* : laissez la valeur *115200* qui est présente par défaut ;
7. *Pulse or Tone* : si votre ligne téléphonique fonctionne à fréquences vocales (ce qui est le cas presque partout en France), sélectionnez *Tone* ; si votre ligne fonctionne avec les impulsions, sélectionnez *Pulse* ;
8. *Phone Number* : rentrez le numéro de téléphone de votre fournisseur d'accès ;
9. *Choose Modem Config Method* : répondez *No* ;
10. *Manually Select Modem Port* : tapez `/dev/modem`, qui est le lien symbolique qui pointe vers le bon périphérique ;
11. *Properties of nom_de_la_connexion* : si vous pensez avoir bien répondu à toutes les questions, sélectionnez *Finished - Write files and return to main menu* et *OK* à l'écran suivant ;

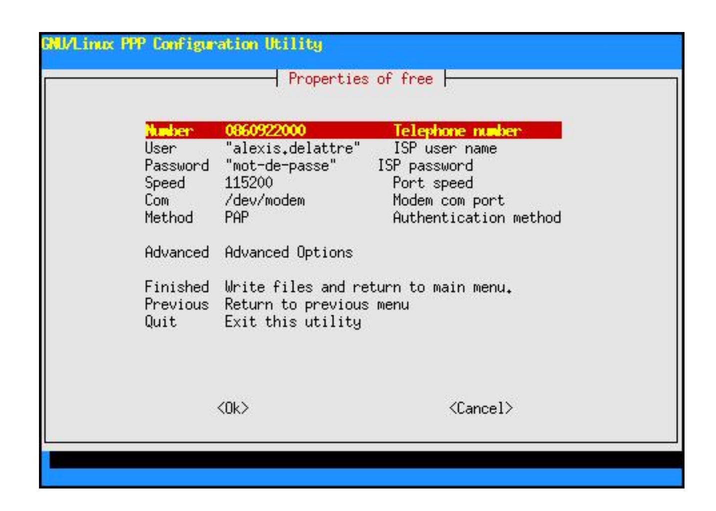


FIGURE 17.2 – pppconfig : propriétés de la connexion

12. *Main Menu* : sélectionnez *Quit - Exit this utility*.

Pour créer une deuxième connexion, changer une connexion existante ou supprimer une connexion, relancez cet assistant et laissez-vous guider par les boîtes de dialogues (qui ne sont malheureusement pas encore traduites).

Se connecter

Pour se connecter au fournisseur d'accès, c'est très simple :

```
# pon nom_de_la_connexion
```

où *nom_de_la_connexion* est le nom que vous aviez entré à la première question de l'assistant.

Vous devez normalement entendre le modem se connecter. Pour suivre l'établissement de la connexion, tapez :

```
# plog -f
```

Dès que vous voyez une ligne du genre :

```
Dec 27 19:42:54 alpy pppd[1825]: Script /etc/ppp/ip-up ↩
    started (pid 1843)
```

cela signifie que la connexion est établie. Vous pouvez alors arrêter l'affichage des messages (encore appelés *logs*) par la combinaison de touches Ctrl-c.

Pour se déconnecter :

```
# poff
```

Pour permettre à un simple utilisateur de se connecter et se déconnecter, il faut le rajouter aux groupes *dialout* et *dip* ; et pour lui permettre d'utiliser la commande **plog**, il faut le rajouter au groupe *adm* :

```
# adduser toto dialout
# adduser toto dip
# adduser toto adm
```

où *toto* est le nom de l'utilisateur à qui vous voulez rajouter les droits. Il pourra alors lancer lui-même les commandes **pon**, **poff** et **plog**.

Chapitre 18

Le système de gestion des paquets Debian

18.1 Généralités

Qu'est-ce qu'un paquet ?

Ceci avait été abordé dans la première partie de cette formation à la section 2.2.

Les trois acteurs de la gestion des paquets

Trois programmes s'occupent de la gestion des paquets Debian : **dpkg**, **aptitude** et **dselect** :

Couche	Programme	Fonction
supérieure	**aptitude** ou **dselect**	Gestion intelligente des paquets : sources, versions, dépendances et conflits
inférieure	**dpkg**	Installation et retrait de paquets

TABLE 18.1 – Les 3 acteurs de la gestion des paquets

18.2 Dpkg

Utilité

Il faut éviter de l'utiliser en temps normal pour installer et désinstaller des paquets, puisqu'il ne gère pas les dépendances entre paquets.

Par contre, c'est souvent le seul moyen d'installer des paquets qui ne sont pas présents dans la distribution. Il faut alors télécharger les fichiers correspondant aux paquets et les installer avec la commande **dpkg**.

Utilisation

Les commandes à savoir sont les suivantes :

- Installe les paquets paquet1 et paquet2 (comme dpkg ne gère pas les dépendances, il faut installer en même temps les paquets qui dépendent l'un de l'autre ; si une ancienne version du paquet est déjà installée, elle sera remplacée) :

```
# dpkg -i paquet1_0.1_i386.deb paquet2_0.2_i386.deb
```

- Désinstalle le paquet paquet1 mais ne supprime pas ses fichiers de configuration :

```
# dpkg -r paquet1
```

- Désinstalle le paquet paquet1 et supprime ses fichiers de configuration :

```
# dpkg -r --purge paquet1
```

- Reconfigure le paquet paquet1 qui est déjà installé :

```
# dpkg-reconfigure paquet1
```

- Donne le nom du paquet qui a installé le fichier /usr/bin/vim (la réponse est facile, c'est le paquet *vim* !) :

```
% dpkg -S /usr/bin/vim
```

- Affiche la liste des fichiers installés par le paquet *vim* :

```
% dpkg -L vim
```

- Affiche la liste des paquets installés :

```
% dpkg -l
```

Pour plus d'informations ou pour avoir la liste complète des options disponibles, consultez le manuel de *dpkg* :

```
% man dpkg
```

18.3 aptitude

Utilité

aptitude est la couche qui apporte une certaine intelligence et une grande facilité d'utilisation au système de gestion des paquets Debian. Pour cela, on définit les sources des paquets dans un fichier de configuration. aptitude gère ensuite l'installation et le retrait des paquets, en tenant compte de leurs dépendances, et s'occupe d'aller chercher les paquets sur vos CD ou DVD, ou de les télécharger s'ils sont sur une source réseau.

aptitude est donc utilisé pour installer et retirer les paquets.

Définir les sources des paquets

La théorie

Les sources des paquets sont définies dans le fichier de configuration `/etc/apt/sources.list`. Une source doit tenir sur une seule ligne (pas de retour à la ligne au milieu de la définition d'une source) et commencer par un des deux mots clés suivants :

– **deb** pour définir une source de paquets binaires ;
– **deb-src** pour définir une source de paquets sources (cela n'intéressera que les développeurs qui veulent examiner le code source des programmes).

Pour ajouter ou retirer une source réseau ou fichier, il faut éditer le fichier « à la main ».

Pour ajouter comme source un CD ou DVD Debian, il faut exécuter la commande suivante :

```
# apt-cdrom add
```

Pour enlever un CD ou DVD Debian de la liste des sources, il faut éditer le fichier et supprimer la ligne correspondant au CD ou au DVD.

Vous trouverez tous les détails sur la syntaxe dans **man sources.list**.

Ajouter les sources de contribution

Si vous disposiez déja d'une connexion à Internet, vous avez déja dû définir les sources de paquets. Mais il ne s'agit que des sources officielles de Debian. Le projet Debian maintient également des dépots pour les paquets qui ne respectent pas strictement les principes du logiciel libre selon Debian. Il s'agit des dépôts *contrib*, contenant des logiciels libres mais basés sur d'autres non libres (c'est souvent le cas des programmes écrits en Java) et *non-free*, contenant des logiciels non libres.

Si vous n'avez pas défini les sources de paquets lors de l'installation, ou si vous souhaitez bénéficier de ces paquets non libres, éditez votre fichier `/etc/apt/sources.list`. Pour définir les dépots utilisés, ajoutez les lignes de définition suivantes. Pour utiliser également les dépots non-libres, rajouter à la fin de chaque dépôt réseau le mot *non-free*. Il doit ressembler à ceci (les dépôts utilisés peuvent être différents) :

```
deb http://ftp.fr.debian.org/debian/ squeeze main [contrib] [ ↩
    non-free]
deb http://security.debian.org/ squeeze/updates main [contrib ↩
    ] [non-free]
...
```

Utilisation

Les commandes à savoir sont les suivantes :

– Met à jour la liste des paquets disponibles (pour les sources réseau ou fichier, il doit aller voir si elles ont été mises à jour, pour les sources CD, il ne fait rien de particulier) :

  ```
  # aptitude update
  ```

– Met à jour tous les paquets déjà installés à la dernière version disponibles dans les sources, sauf ceux qui nécessitent d'en désinstaller d'autres :

```
# aptitude safe-upgrade
```

– Idem que la commande précédente, mais cette commande effectue également des désinstallations des dépendances qui ont pu changer (par exemple lors d'un passage de la version stable à la version instable de Debian) :

```
# aptitude full-upgrade
```

– Installe les paquets `paquet1` et `paquet2` et tous les paquets dont ils dépendent :

```
# aptitude install paquet1 paquet2
```

– Désinstalle le paquet `paquet1` sans effacer ses fichiers de configuration :

```
# aptitude remove paquet1
```

– Idem que la commande précédente mais ses fichiers de configuration sont supprimés :

```
# aptitude purge paquet1
```

– Efface du disque dur les paquets téléchargés pour être installés (inutile quand la source est un CD ou un fichier du système de fichiers local) :

```
# aptitude clean
```

Pour plus d'informations ou pour avoir la liste complète des options disponibles, consultez le manuel d'*aptitude* :

```
% man aptitude
```

Rechercher un paquet et afficher sa description

Pour chercher un paquet dont le nom ou la description contient certains mots, utilisez la commande suivante :

```
% apt-cache search liste_de_mots_clés
```

Par exemple, pour chercher un compilateur Fortran, tapez :

```
% apt-cache search fortran compiler
```

Il recherche alors les termes *fortran* et *compiler* dans la description de tous les paquets disponibles. Dans les réponses, vous obtenez notamment :

```
gfortran - The GNU Fortran 95 compiler.
```

Pour afficher les caractéristiques et la description du paquet *paquet1*, utilisez la commande suivante :

```
% aptitude show paquet1
```

Par exemple :

```
% aptitude show gfortran

Paquet~: gfortran
État: non installé
Version~: 4:4.3.1-2
Priorité~: optionnel
Section~: devel
Responsable~: Debian GCC Maintainers <debian-gcc@lists.debian ↩
    .org>
Taille décompressée~: 41,0k
Dépend: cpp (>= 4:4.3.1-2), gcc (>= 4:4.3.1-2), gfortran-4.3  ↩
    (>= 4.3.1-1)
Suggère: gfortran-multilib, gfortran-doc
Fournit: fortran-compiler
 Description~: The GNU Fortran 95 compiler
 This is the GNU Fortran 95 compiler, which compiles Fortran  ↩
    95 on platforms
 supported by the gcc compiler. It uses the gcc backend to  ↩
    generate optimized
 code.

 This is a dependency package providing the default GNU  ↩
    Fortran 95 compiler.

Étiquettes: devel::compiler, devel::lang:fortran, role::dummy ↩
    , suite::gnu
```

18.4 Dselect

Dselect est une alternative à *aptitude* pour la gestion intelligente des paquets.

Il a un certain nombre d'avantages par rapport à *aptitude*, mais il a aussi de nombreux inconvénients, notamment la complexité d'utilisation pour un débutant ainsi que la difficulté de résoudre les problèmes de dépendance.

18.5 Avant d'aller plus loin

Mettre à jour son système

Des problèmes de sécurité sont régulièrement découverts dans les logiciels, et sont rapidement corrigés. Il est donc possible que des mises à jour importantes soient disponibles. Pour les installer, tapez :

```
# aptitude update
# aptitude safe-upgrade
```

Configurer… l'outil de configuration Debian !

Debconf est l'outil de configuration des paquets Debian. Quand vous installerez de nouveaux logiciels qui nécessitent une configuration, il vous posera quelques questions.

Par défaut, Debconf est réglé pour un débutant qui ne souhaite pas se poser trop de questions. Nous allons donc le régler de façon un peu plus curieuse :

```
# dpkg-reconfigure debconf
```

Il vous demande alors quelle interface vous souhaitez utiliser : conservez l'interface *Dialogue*. Quant à la priorité des questions, choisissez *élevée*.

Compléter l'installation de Vim

Maintenant que, vous avez accès à l'intégralité des paquets, vous allez pouvoir compléter l'installation de Vim, en installant le programme *par*, qui permet de reformater du texte dans Vim :

```
# aptitude install par
```

Ce programme apporte deux nouvelles fonctions à vim quand vous êtes en mode commande :

- la touche **#** coupe les lignes d'un paragraphe à 72 caractères (on dit *wrapper*), ce qui est la norme pour les documents texte que vous envoyez (mail, post dans les news...) ;
- la touche @ fait la même chose que **#** mais en justifiant le texte.

Installer un nouveau *pager* : most

Le *pager* est un programme qui sert à afficher du texte. Vous avez le choix entre plusieurs programmes, grâce à un mécanisme d'*alternatives*. Je vous propose d'installer et d'utiliser *most*, un pager qui permet entre autres de colorer les pages de manuel.

```
# aptitude install most
# update-alternatives --config pager
```

La seconde commande vous demande de choisir le pager à utiliser. Répondez en donnant le numéro qui correspond à *most*. Maintenant, c'est ce programme qui sera utilisé pour afficher les manuels.

Chapitre 19

Configurer le shell

19.1 Qu'est-ce qu'un shell ?

Le shell est ce qui s'exécute quand vous vous logguez. C'est lui qui vous présente le prompt, qui envoie vos commandes au système, qui enregistre certaines variables. Il est encore là quand vous vous délogguez.

Vous avez un vaste choix de shells différents. Le shell par défaut sous Debian s'appelle bash. Mais il existe aussi csh, tcsh, zsh, sash...

Je vais vous proposer d'installer le shell Zsh, de télécharger des fichiers de configuration et ensuite de passer de *bash* à *zsh*.

19.2 Installer et configurer Zsh

Installer le paquet et les fichiers de configuration

Installez le paquet *zsh*, qui contient le shell du même nom :

```
# aptitude install zsh
```

Remplacez les fichiers de configuration par défaut par mes fichiers de configuration :

```
# cd ~/fichiers-config
# cp zshrc zshenv zlogin zlogout /etc/zsh/
# cp dir_colors /etc/
```

ou, si vous ne suivez pas ma formation depuis le début :

```
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
    zshrc
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
    zshenv
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
    zlogin
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
    zlogout
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
    dir_colors
```

```
# mv zshrc zshenv zlogin zlogout /etc/zsh/
# mv dir_colors /etc/
```

Configurer le proxy

Si vous devez passer par un proxy pour accéder à Internet, au lieu de taper à chaque fois **export http_proxy=...**, vous allez éditer en root le fichier `/etc/zsh/zshenv` puis décommenter et personnaliser les lignes adéquates :

– si votre proxy ne requiert pas d'authentification par login et mot de passe :

```
# Proxy HTTP / FTP sans mot de passe
export http_proxy="http://proxy.exemple.org:8080"
export ftp_proxy="ftp://proxy.exemple.org:8080"

# Ne pas passer par le proxy pour les domaines locaux
export no_proxy="exemple.org"
```

où *proxy.exemple.org* est le nom DNS de votre proxy et *8080* son port.

– si votre proxy requiert une authentification par login et mot de passe :

```
# Proxy HTTP / FTP avec mot de passe
export http_proxy="http://login:password@proxy.exemple. ↩
    org:8080"
export ftp_proxy="ftp://login:password@proxy.exemple.org ↩
    :8080"

# Ne pas passer par le proxy pour les domaines locaux
export no_proxy="exemple.org"
```

où *proxy.exemple.org* est le nom DNS de votre proxy, *8080* son port, *login* et *password* votre login et mot de passe pour le proxy.

Enregistrez et quittez.

Changer de Shell

Pour changer de shell, un utilisateur doit exécuter la commande **chsh** et préciser où se trouve son nouveau shell. Il bénéficiera alors des fichiers de configuration par défaut que vous venez d'installer. Il pourra aussi mettre ses propres fichiers de configuration dans son home (même nom mais précédé d'un point).

Pour passer à Zsh, un utilisateur doit donc taper :

```
% chsh
Enter the new value, or press return for the default
    Login Shell [/bin/bash]:/bin/zsh
```

Pour que le changement soit effectif, il faut se déconnecter (Ctrl-d) et se relogguer. Vous pouvez alors admirer la différence (Figure 19.1) !

```
formation@debian:~$ echo "Voici Bash, le shell par défaut de Debian"
Voici Bash, le shell par défaut de Debian
formation@debian:~$ zsh
21:39 formation@debian ~% echo "Et voici Zsh, un shell qui gère"
Et voici Zsh, un shell qui gère
21:39 formation@debian ~%
```

FIGURE 19.1 – Bash et Zsh

Les informations sur les comptes utilisateur sont en fait stockées dans le fichier `/etc/passwd`. Ce fichier se présente ainsi :

```
<login>:<hash du mot de passe (optionnel)>:<uid>:<gid>:<nom ↩
    complet>:<répertoire personnel>:<shell>
```

soit par exemple pour moi :

```
jpountz:x:1000:1000:Adrien Grand,,,:/home/jpountz:/bin/zsh
```

Vous trouverez plus de détails sur le fichier `/etc/passwd` grâce à sa page de manuel :

```
% man 5 passwd
```

Lors d'un changement de shell, la commande **chsh** se charge donc de modifier l'information correspondant au shell utilisateur.

ASTUCE

Zsh propose une *autocomplétion* très efficace. Il s'agit d'un outil qui permet d'accélérer la saisie des commandes, en les complétant automatiquement dès qu'il n'y a plus d'ambigüité. Pour l'utiliser, il faut appuyer sur la touche **Tab** pendant la frappe, mais essayez plutôt :

```
% aptiTab insTab vimTab
```

Shell par défaut pour les nouveaux utilisateurs

Pour changer le shell par défaut pour les nouveaux utilisateurs, il faut modifier le fichier de configuration de la commande **adduser** qui sert à ajouter un utilisateur au système. En root, éditez le fichier `/etc/adduser.conf` avec *vim* :

```
# vim /etc/adduser.conf
```

Changez la ligne :

```
DSHELL=/bin/bash
```

par la ligne :

```
DSHELL=/bin/zsh
```

Enregistrez et quittez. Comme ça, quand le root rajoutera un nouvel utilisateur avec la commande :

```
# adduser toto
```

ce nouvel utilisateur aura un shell *zsh* bien configuré.

Qu’est-ce que le *PATH* ?

PATH est une variable d’environnement. Pour afficher le contenu d’une variable d’environnement, on utilise la commande **echo** :

```
% echo $PATH
/usr/local/bin:/usr/local/sbin:/bin:/usr/bin:/usr/sbin:/usr/ ↩
    bin/X11:/usr/X11R6/bin:/usr/games:
/sbin:/home/alexis/bin
```

La variable *PATH* contient la liste de tous les répertoires dans lesquels le système va chercher les exécutables des commandes que vous tapez au prompt, séparés par des « : ». Par exemple, le répertoire `/bin/` contient les commandes Unix de base, et vous pouvez vérifier qu’il est bien dans le *PATH*.

Pour modifier le *PATH*, éditez le fichier de configuration `/etc/zsh/zshenv` et ajoutez ou supprimez un répertoire à la ligne qui commence par *export PATH=*.

Chapitre 20

Utiliser des médias de stockage

Comme expliqué au Chapitre 10, vos fichiers sont écrits dans des systèmes de fichiers, eux-mêmes stockés sur des périphériques de stockage : disquette, clef USB, CD, partition de disque dur... Pour utiliser un support de stockage, on le monte dans l'arborescence.

20.1 /etc/fstab

Le fichier de configuration `/etc/fstab` contient les informations statiques sur le montage des systèmes de fichiers que vous utilisez régulièrement.

La syntaxe du fichier

Les règles de syntaxe du fichier sont les suivantes : une ligne par système de fichier, chaque ligne devant contenir dans l'ordre les informations suivantes séparées par au moins un espace :

1. l'emplacement physique du système de fichiers : `/dev/partition` pour une partition physique,
2. le point de montage (le répertoire doit déjà exister, sinon il faut le créer au préalable avec la commande **mkdir**),
3. le (ou les) type de système de fichiers (par exemple **swap**, **ext3**, **vfat**, **ntfs**, **nfs**, **iso9660** et **udf** pour les CD et DVD), **auto** pour autodétecter le type ;
4. les options de montage, séparées par des virgules :
   - **ro** pour monter le système de fichiers en lecture seule,
   - **rw** pour monter le système de fichiers en lecture-écriture,
   - **noauto** pour que le système de fichiers ne soit pas monté au démarrage (option contraire : **auto**),
   - **user** pour qu'un simple utilisateur puisse monter et démonter le système de fichiers et pas seulement le root (option contraire : **nouser**),
   - **exec** pour permettre l'exécution de binaires (option contraire : **noexec**),
   - **uid**, **gid** et **umask** pour définir des permissions pour l'ensemble du système de fichiers (pour les systèmes déficients comme FAT ou NTFS),
   - **defaults** pour les options par défaut (notamment rw, exec, auto et nouser),
   - et enfin **sw** pour les systèmes de *swap*.

5. la valeur **1** si le système de fichier doit être sauvegardé ou la valeur **0** sinon (mettez **0** si vous n'avez pas de système de sauvegarde),
6. la priorité pour la vérification des systèmes de fichiers par **fsck** au démarrage quand cela est nécessaire : la partition racine doit avoir la plus grande priorité (valeur **1**), les autres doivent avoir une priorité inférieure (valeur **2**). Les systèmes de fichiers qui ne doivent pas être vérifiés auront la valeur **0**.

Donc pour un système classique, le fichier contient par exemple :

```
/dev/hda1  /              ext3        defaults          0  ↩
        1   # partition de DD
/dev/hda2  none           swap        sw                0  ↩
        0   # partition de DD
/dev/hda5  /home          ext3        defaults          0  ↩
        2   # partition de DD
proc       /proc          proc        defaults          0  ↩
        0   # SF virtuel
/dev/fd0   /media/floppy0 auto        user,noauto       0  ↩
        0   # disquette
/dev/sda1  /media/clef0   vfat        user,noauto       0  ↩
        0   # clef USB
/dev/hdc   /media/cdrom0  udf,iso9660 ro,user,noauto    0  ↩
        0   # CD ou DVD
```

Monter et démonter une partition citée dans `fstab`

Un des avantages d'utiliser le fichier `/etc/fstab` est que le montage et le démontage des systèmes de fichiers cités dans ce fichier de configuration sont très simples. Il suffit d'utiliser la commande **mount** pour monter et **umount** pour démonter, suivie du périphérique *ou* du répertoire de montage.

Ainsi, les deux commandes suivantes sont équivalentes, et permettent de monter le CD présent dans le lecteur :

```
# mount /media/cdrom0
```

```
# mount /dev/hdc
```

Comme nous avons précisé l'option *user* pour le lecteur de CD dans `fstab`, les deux commandes précédentes peuvent être exécutées en tant que simple utilisateur. Dans ce cas, seul l'utilisateur en question et le root pourront démonter le système de fichiers avec l'une des deux commandes suivantes :

```
# umount /media/cdrom0
```

```
# umount /dev/hdc
```

Important

Pour démonter un système de fichier, il faut qu'il ne soit plus utilisé ni parcouru par aucun utilisateur ni aucun processus. Sinon, la commande **umount** renverra le message d'erreur suivant :

```
umount : /media/cdrom0 : périphérique occupé
```

Pour voir qui est responsable de cette occupation, utilisez la commande **fuser** qui liste les fichiers ouverts ainsi que les noms des processus, et les utilisateur qui les ont lancés :

```
# aptitude install psmisc
% fuser -vm /media/cdrom0
```

20.2 Monter ses partitions Windows

Préparer le montage d'une partition FAT

Le système de fichiers de type FAT (FAT 16 ou FAT 32) est utilisé par Windows 95/98/ME et parfois par Windows 2000/XP. Le pilote Linux pour ce type de système de fichiers permet d'y avoir accès en lecture et en écriture.

Supposons que votre partition Windows de type FAT soit `/dev/hda1` (première partition primaire sur le disque dur IDE maître de la première nappe). Nous allons la monter dans le répertoire `/media/win1` qu'il faut créer au préalable :

```
# mkdir /media/win1
```

Ensuite, éditez en root le fichier `/etc/fstab` et rajoutez la ligne suivante :

```
/dev/hda1      /media/win1       vfat       defaults,user  ↩
        0      0
```

Préparer le montage d'une partition NTFS

Le système de fichiers de type NTFS est souvent utilisé par Windows 2000, XP et Vista. Pour pouvoir lire et écrire sur ces systèmes, il faut d'abord installer un pilote particulier, NTFS-3g.

Note

Le noyau Linux comporte déjà un pilote pour le système de fichiers NTFS, mais celui-ci ne permet pas de créer de nouveaux fichiers. NTFS-3g, qui est un pilote en espace utilisateur, c'est à dire exécuté en dehors du noyau, permet un accès complet en lecture et en écriture.

```
# aptitude install ntfs-3g
```

Supposons que votre partition Windows de type NTFS soit `/dev/sda5` (premier lecteur logique sur le premier disque dur SATA). Nous allons la monter dans le répertoire `/media/win2` qu'il faut créer au préalable :

```
# mkdir /media/win2
```

Ensuite, éditez en root le fichier `/etc/fstab` et rajoutez la ligne suivante :

```
/dev/sda5     /media/win2       ntfs-3g    defaults,user  ↩
        0     0
```

Monter les partitions

Vous avez rajouté les entrées nécessaires dans le fichier `/etc/fstab` : vos partitions Windows seront donc dorénavant montées automatiquement dès le démarrage. Mais pour éviter de redémarrer, vous allez simplement demander au système de monter les partitions citées dans fstab et non déjà montées avec la commande suivante :

```
# mount -a
```

Si aucun message d'erreur n'apparaît, vous devez maintenant pouvoir voir le contenu de votre ou vos partition(s) Windows dans l'arborescence de votre système.

Modifier les droits sur les partitions Windows

Par défaut, les partitions Windows montées appartiennent à root, et dans le cas des systèmes de fichiers NTFS, elles ne sont pas lisibles par les autres utilisateurs. Pour modifier les droits d'accès appliquées aux partitions Windows, vous pouvez rajouter des options dans la ligne qui leur correspond dans le fichier `/etc/fstab`.

Par exemple, si vous voulez que sur la partition Windows `/dev/hda1` formatée en FAT, les fichiers et les répertoires :

- appartiennent à root, dont l'ID est **0**,
- appartiennent à un groupe win, que vous avez créé avec la commande **addgroup**, et dont l'ID est 1003,
- aient des droits **rw-rw-r--** pour les fichiers réguliers, et **rwxrwxr-x** pour les répertoires,

alors la ligne correspondant à la partition dans `fstab` devient la suivante :

```
/dev/hda1     /media/win1    vfat     defaults,user,uid=root, ↩
    gid=win,umask=113,dmask=002    0     0
```

Pour que les changements soient pris en compte, la commande **mount -a** ne suffit pas. Il faut démonter et remonter la partition :

```
% umount /media/win1
% mount /media/win1
```

20.3 Monter sa clé USB

Si vous avez une clé USB (ou n'importe quel périphérique compatible avec la norme de stockage de masse USB), commencez par créer le répertoire dans lequel vous monterez la clé :

```
# mkdir /media/clef0
```

Identifiez le nom de périphérique correspondant à votre clef USB : afichez la liste des partitions disponibles avec la commande **cat /proc/partitions**, puis introduisez votre clef, et, après quelques secondes, affichez à nouveau le contenu de `/proc/partitions`, dans lequel votre clef à dû apparaître (chez moi, elle s'appelle `/dev/sda1`, mais si vous avez des disques SATA, ce nom peut être décalé de quelques lettres).

Enfin, si votre clé USB est formatée en FAT et que vous voulez que les fichiers une fois montés appartiennent à root et au groupe win (cf. plus haut), rajoutez la ligne suivante à la fin du fichier `/etc/fstab` :

```
/dev/sda1    /media/clef0    vfat    defaults,user,uid=win, ↩
    gid=win,umask=113,dmask=002    0    0
```

Vous pouvez alors monter votre clé USB :

```
% mount /media/clef0
```

Attention

N'oubliez pas de démonter votre clé USB avant de la débrancher, sous peine de corrompre les données qui y sont stockées :

```
% umount /media/clef0
```

Chapitre 21

Le réseau et la sécurité

21.1 Introduction à la sécurité

Qui est concerné ?

Ce chapitre vous concerne si votre ordinateur n'est pas isolé mais connecté à un réseau local ou à Internet.

Mais pourquoi s'embêter ?

Le raisonnement de base est le suivant : *La sécurité de ma machine, je m'en fous : y'a rien de précieux sur ma machine... personne n'a intérêt à me pirater !*

AVERTISSEMENT
C'est *FAUX* ! Les pirates recherchent les machines vulnérables pour avoir accès à un compte sur ces machines. Ils peuvent ainsi lancer leur vraie attaque destructrice depuis cette machine vulnérable au lieu de le faire depuis leur machine personnelle. Ainsi, on remonte beaucoup plus difficilement jusqu'à eux !

Autre raisonnement dangereux : *J'ai Linux, donc je suis tranquille niveau sécurité !*

AVERTISSEMENT
C'est encore une fois *FAUX* ! Il y a des failles de sécurité, même sous Linux. Par exemple, sur les noyaux 2.6.2 et inférieurs, une faille permet à n'importe quel utilisateur de devenir root ! On appelle ça un *local root exploit*. Plus grave, il y a régulièrement des failles dans des programmes qui permettent à un pirate d'exécuter du code sur la machine avec les mêmes privilèges que l'application vulnérable ! On appelle ça un *remote exploit* ; et quand l'application vulnérable tourne en root (c'est le cas du serveur d'accès à distance `SSH` par exemple), alors on appelle ça un *remote root exploit*, et le pirate a alors le contrôle total sur la machine !

Pour l'avoir *vécu*, se faire pirater son système n'a rien d'agréable, et le seul moyen de se retrouver en sécurité consiste à le réinstaller totalement et à changer tous ses mots de passe.

Morale

J'espère que je vous ai convaincu de l'importance de se tenir au courant des problèmes de sécurité et de mettre votre système à jour dès qu'une faille est découverte et réparée.

L'avantage d'appartenir au monde du logiciel libre est que tous les programmeurs du monde entier ont accès au code source du noyau et des programmes et peuvent alors corriger les failles de sécurité. La correction des failles est donc beaucoup plus rapide qu'avec d'autres systèmes d'exploitation non libres.

21.2 Protéger son système

Les failles de sécurité dans les paquets Debian

Avec Debian, quand un paquet a une faille de sécurité, une équipe spéciale, la *security team*, se charge de mettre rapidement à disposition des utilisateurs une version corrigée du paquet contenant le programme vulnérable sur un site dédié.

Pour être mis au courant de la disponibilité d'une mise à jour de sécurité, il faut s'abonner à la liste de distribution *debian-security-announce*. Pour s'inscrire, il suffit de se rendre à l'adresse http://www.debian.org/MailingLists/subscribe. Par la même occasion, vous pouvez vous abonner à la liste *debian-announce* pour recevoir les annonces des sorties de nouvelles versions de la distribution Debian. Je vous conseille de vous abonner également à la liste *debian-news* pour recevoir chaque semaine un résumé de l'actualité du projet Debian.

Quand une faille de sécurité est corrigée par Debian, vous recevez un message par la liste de distribution *debian-security-announce*. Ce message décrit la faille et la procédure pour mettre à jour facilement votre système.

En pratique, la procédure de mise à jour est toujours la même. Normalement, l'installateur de Debian a dû placer dans la liste des sources de paquets `/etc/apt/sources.list` une ligne comme cell-ci :

```
deb http://security.debian.org/ lenny/updates main contrib ↩
    non-free
```

Ensuite, il suffit de mettre à jour la liste des paquets puis les paquets eux-mêmes :

```
# aptitude update
# aptitude safe-upgrade
```

Les failles de sécurité noyau

Il arrive également qu'il y ait des failles de sécurité dans le noyau Linux. L'équipe de développement du noyau se charge alors de corriger la faille au plus vite.

Il n'existe pas à ma connaissance de liste d'annonce officielle pour être mis au courant des failles de sécurité du noyau... mais il suffit de jeter un oeil régulièrement aux sites d'actualité Linux, comme par exemple LinuxFr, qui relayent ce genre d'informations.

L'équipe de sécurité de Debian s'occupe également de surveiller les failles du noyau. À moins que vous ne choisisssiez de compiler et d'utiliser votre propre noyau, vous pourrez donc appliquer les mises à jour du noyau de la même façon que celle des

autres paquets. Après une mise à jour du noyau, il est nécessaire de redémarrer pour utiliser le nouveau noyau corrigé.

Surveiller son système en lisant les logs

Les logs sont des fichiers textes produits par le système, dans lesquels celui-ci raconte ce qu'il fait et ce qui lui arrive. Il donne des renseignements sur ce que font les programmes, les connexions qui arrivent à votre machine, les personnes qui s'y connectent.

Les logs se trouvent dans le répertoire `/var/log/`. Il faut appartenir au groupe *adm* pour pouvoir les lire. Rajoutez donc votre compte utilisateur à ce groupe pour éviter de lire les logs en root :

```
# adduser toto adm
```

Les fichiers de logs les plus importants sont :

– `syslog` : c'est le fichier de log principal. Il contient tous les messages du noyau (que l'on retrouve dans `kernel.log`), tous les messages des serveurs (que l'on retrouve dans `daemon.log`), tous les messages de la cron...
– `auth.log` : il vous raconte tout ce qui concerne les authentifications.

Lire régulièrement les logs de sa machine permet de voir si quelqu'un essaye de vous attaquer. Cela permet aussi de voir si tout se passe bien au niveau du système, du noyau, etc.

Rajouter une console de logs

Il peut être intéressant d'avoir une console sur laquelle les logs défilent *en direct*. Cela permet de voir en temps réel ce qui se passe au niveau du système, et donc de résoudre les éventuels problèmes plus rapidement.

Pour cela, éditez en root le fichier de configuration de *syslog* (le programme qui gère les logs) `/etc/rsyslog.conf`. Décommentez les 4 lignes à l'endroit où les commentaires parlent de cette fonction (vers la ligne 50) :

```
daemon,mail.*;\
        news.=crit;news.=err;news.=notice;\
        *.=debug;*.=info;\
        *.=notice;*.=warn       /dev/tty8
```

Pour que le système tienne compte de cette modification, tapez :

```
# /etc/init.d/rsyslog restart
```

En allant sur la console n°8, vous devez déjà voir une première ligne de texte qui vous informe que *syslog* a redémarré !

Aller plus loin...

Pour en savoir plus sur l'art et la manière de sécuriser un système Debian, je vous conseille la lecture du Manuel de sécurisation de Debian.

Chapitre 22

Le Web et le FTP en console

22.1 Surfer sur le web en console ?

C'est possible... mais pas très joli (Figure 22.1) ! Il existe (au moins) trois navigateurs en mode texte : lynx, w3m et links2 qui se trouvent dans les paquets du même nom.

Si, comme je vous l'avais conseillé, vous n'avez imprimé cette formation que jusqu'à ce point, vous pouvez maintenant suivre ma formation en ligne en lançant w3m, qui est normalement déjà installé :

```
% w3m http://formation-debian.via.ecp.fr/
```

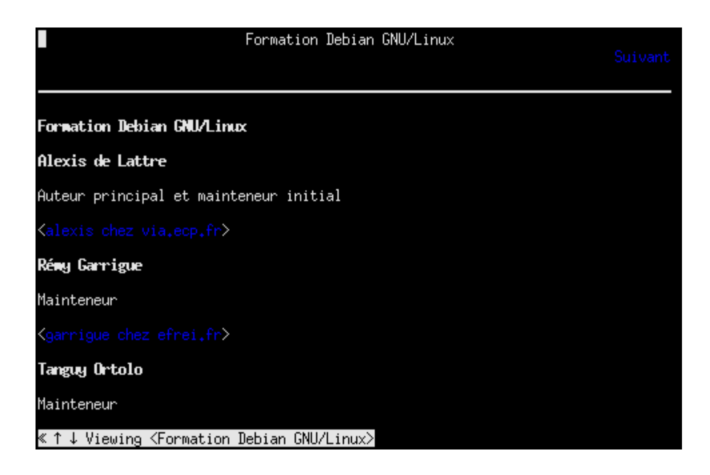


FIGURE 22.1 – w3m

Vivement l'installation du serveur graphique pour pouvoir surfer avec Iceweasel (nom de Firefox sous Debian) !

22.2 Le FTP en console

Le client FTP le plus facile à utiliser en console est, à mon goût, lftp (Figure 22.2).

```
16:54 tanguy@glamdring ~% lftp sftp://tanguy@localhost
Mot de passe :
lftp tanguy@localhost ~> ls debian/formation-debian
.svn/  branches/  tags/  trunk/
lftp tanguy@localhost ~> cd debian/formation-debian/trunk
lftp tanguy@localhost ~/debian/formation-debian/trunk> ls
.cvsignore  Makefile.lenny    fichiers-config.tar.gz  src/
.svn/       TODO              html/                   tex/
COPYRIGHT   WELCOME           pdf/                    txt/
Makefile    fichiers-config/  rtf/                    utils/
lftp tanguy@localhost ~/debian/formation-debian/trunk>
```

FIGURE 22.2 – lftp (attention les yeux !)

Installation et configuration

Commencez par installer le paquet :

```
# aptitude install lftp
```

Ensuite, installez mon fichier de configuration pour lftp :

```
# cp ~/fichiers-config/lftp.conf /etc/
```

ou :

```
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
    lftp.conf
# mv lftp.conf /etc/
```

Utiliser lftp

L'utilisation de lftp est basée sur des commandes :

1. Pour se connecter :
   – en utilisateur *toto* sur le serveur *ftp.exemple.org* :

   ```
   % lftp ftp://toto@ftp.exemple.org
   ```

   – en anonyme sur le serveur FTP *archive.debian.org* :

   ```
   % lftp ftp://archive.debian.org
   ```

2. Une fois connecté, un nouveau prompt apparaît :

   ```
   lftp login@nom_du_serveur ~>
   ```

   Les commandes de base disponibles à ce prompt sont les suivantes (la complétion automatique des noms de fichiers marche) :
   – **help** : affiche la liste des commandes disponibles,
   – **help commande** : affiche l'aide de la commande,
   – **ls** : liste le contenu du répertoire distant,
   – **ls -la** : liste le contenu du répertoire distant avec les fichiers cachés et les permissions,
   – **cd répertoire** : change de répertoire distant,
   – **lcd répertoire** : change de répertoire local,
   – **get fichier** : télécharge le fichier,

- **get *.img** : télécharge tous les fichiers avec l'extension *img*,
- **mirror répertoire** : télécharge le répertoire,
- **put fichier** : dépose le fichier,
- **put test*** : dépose tous les fichiers dont le nom commence par *test*,
- **exit** : met fin à la connexion.

Chapitre 23

Configurer son serveur de mail local

Les ordinateurs sous GNU/Linux ont besoin d'un serveur de mail pour fonctionner correctement. Certains logiciels communiquent notamment avec l'administrateur en lui écrivant. Par contre, pour mettre en place un *vrai* serveur de mail destiné à recevoir du courrier de l'extérieur, il faut des entrées dans la DNS et surtout une machine en état de fonctionnement 24h/24 et 7j/7.

Ce chapitre explique seulement la configuration d'un serveur mail qui gère le mail en local et l'envoi des mails. La configuration d'un *vrai* serveur de mail avec les entrées DNS qui vont avec est expliquée dans l'annexe correspondante.

23.1 Installation de Postfix

Je vous propose d'installer Postfix, qui est réputé fiable et facile à configurer :

```
# aptitude install postfix
```

23.2 Configuration de Postfix

Lors de l'installation du paquet, il vous pose des questions de configuration. À la première question *Type de configuration ?*, répondez *Internet avec un smarthost*.

Il vous pose ensuite plusieurs questions :

– Nom de courrier : répondez comme vous le souhaitez.
– Serveur relai SMTP : donnez le nom du serveur SMTP de votre fournisseur d'accès (en général, il est de la forme *smtp.fournisseur*.

Postfix est un logiciel très puissant et sa configuration sort du cadre de ce document. En revanche, Traduc.org propose une traduction très à jour de la documentation officielle de Postfix.

Troisième partie

Debian GNU/Linux en mode graphique

La première partie de cette formation vous a expliqué en détail la procédure d'installation ; et la deuxième partie vous a appris les commandes et les outils de base. Cette troisième partie va vous apprendre à installer un serveur graphique : vous pourrez enfin avoir un système qui ressemble à un poste de travail classique avec un bureau et des icônes ! Pour ceux dont l'objectif était d'installer un système serveur, cette partie ne leur sera pas utile, puisque les serveurs n'ont généralement pas besoin de serveur graphique... par contre, la quatrième partie IV est faite pour eux !

Les notions de serveur graphique, de bureau et de gestionnaire de fenêtre

- Le *serveur graphique* est le programme qui permet de passer en « mode graphique » en utilisant les fonctions avancées de la carte graphique. Il gère notamment le clavier, la souris, les polices de caractères, l'écran (résolution, nombre de couleurs, etc.) et la carte graphique, et fournit aux programmes qui le demandent un affichage et un contrôle. Sous Debian, c'est le serveur X.Org qui est utilisé.
- Le *bureau* est le programme qui s'occupe d'afficher un menu, une barre de lancement, une barre des tâches, des icônes sur le bureau, etc... Il existe également de nombreux bureaux sous Linux, mais les deux plus connus sont GNOME et KDE.
- Le *gestionnaire de fenêtre* s'occupe de « décorer » les fenêtres, en leur ajoutant des bordures, une barre de titre, et des boutons de contrôle pour pouvoir les réduire, les fermer ou les agrandir. Les bureaux graphiques fournissent normalement leur propre gestionnaire de fenêtres : celui de GNOME s'appelle Metacity, mais il est possible d'en utiliser une autre.

Avoir un serveur graphique et un gestionnaire de fenêtre est obligatoire pour travailler « en mode graphique » ; mais on n'est pas obligé d'avoir un bureau ! Certains gestionnaires de fenêtre font aussi office de bureau minimaliste (Windows Maker affiche par exemple un menu de lancement rapide). Normalement, l'utilisateur peut choisir n'importe quelle combinaison de gestionnaire de fenêtres et de bureau, mais certains bureaux recommandent un gestionnaire de fenêtres particulier.

Note

Dans cette formation, nous documentons GNOME, dont vous pouvez voir une capture à la Figure 23.1. Pour les débutants qui suivent cette formation, nous leurs conseillons de suivre nos choix, et, quand ils auront plus d'expérience avec GNU/Linux en général et Debian en particulier, ils pourront essayer d'autres bureaux et d'autres gestionnaires de fenêtres et choisir celui qui leur convient le mieux !

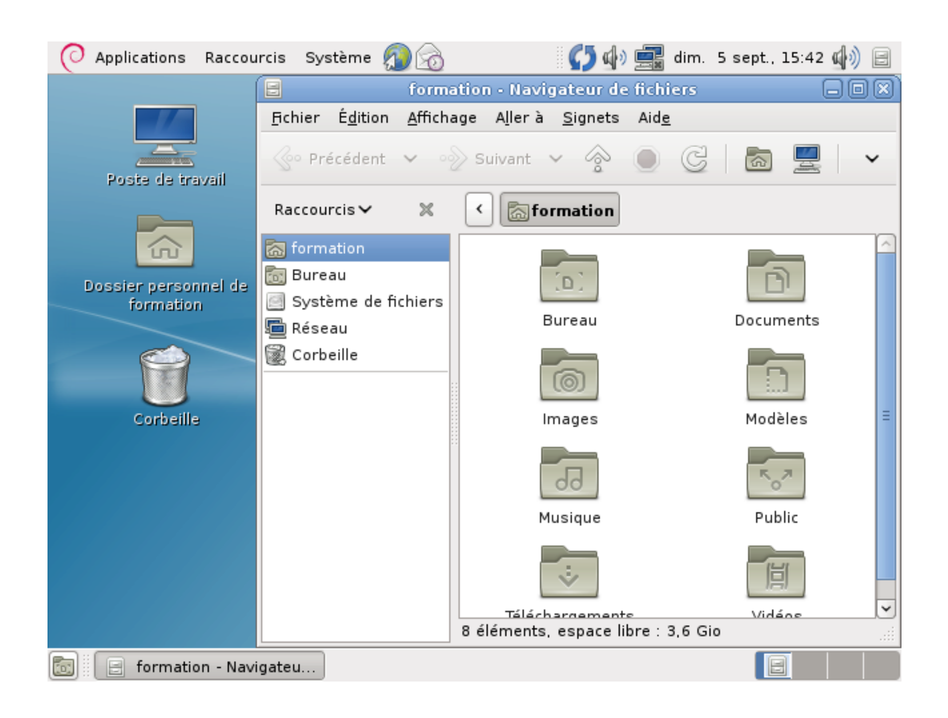


FIGURE 23.1 – Un bureau GNOME

Chapitre 24

Installer le serveur graphique

24.1 Installer le serveur graphique et le bureau GNOME

Installer le serveur graphique

Installez les paquets nécessaires :

```
# aptitude install xorg mesa-utils
```

La denière version du serveur graphique X.Org apporte une fonctionnalité intéressante : il est maintenant capable de déterminer automatiquement une configuration adaptée à votre matériel. Par conséquent, sauf pour des besoins très particuliers, il est inutile de le configurer à la main.

Il est malgré tout possible que vous souhaitiez ajuster quelques paramètres, comme par exemple la sensibilité de la souris. Voici donc des explications sur les concepts utilisés dans la configuration du serveur graphique.

Installer le bureau et le gestionnaire de fenêtre

Le bureau GNOME est constitué de nombreux logiciels. Tous ne vous seront pas nécessairement utiles, mais nous vous proposons d'installer la sélection proposée par Debian, qui occupe à peu près 1 Go, et vous permettra de découvrir de nombreuses possibilités auxquelles vous n'aurez pas forcément pensé. En fait, nous allons exécuter à nouveau **tasksel**, l'outil de sélection de « tâches » que vous avez déjà vu vers la fin de l'installation :

```
# tasksel install desktop
```

Note

Les paquets à charger pèsent environ 500 Mo (ils sont compressés), donc si vous utilisez un miroir réseau, cela peut prendre un moment. Si vous préférez vous contenter d'un bureau GNOME minimal, vous pouvez installer à la place le paquet *gnome-core* :

```
# aptitude install gnome-core
```

Lancer X !

Note

Par la suite, le système graphique se lancera tout seul au démarrage, et il vous sera inutile de le lancer à la main.

Maintenant que tout est prêt, vous allez pouvoir lancer le serveur X en tant que simple utilisateur (on ne lance jamais un server graphique en root) avec la commande suivante :

```
% startx
```

Note

Ce qui suit est donné à titre informatif : si vous êtes pressé, vous pouvez passer au chapitre suivant.

24.2 Manipulations de base

Zapper entre les consoles et le serveur graphique

– Pour passer du serveur graphique à la console numéro *X*, utilisez la combinaison de touches Ctrl-Alt-FX.
– Pour passer de la console *X* à la console *Y*, utilisez la combinaison habituelle Alt-FY.
– Pour revenir sur le serveur graphique, utilisez la combinaison de touches Alt-F7.
– Pour tuer le serveur graphique, s'il ne fonctionne plus correctement, utilisez la combinaison de touches Ctrl-Alt-Retour arrière (la touche au-dessus d'**Entrée**).

24.3 La configuration d'un serveur graphique

Si vous souhaitez peaufiner votre configuration, il vous faut tout d'abord vous renseigner sur votre matériel :
– votre clavier,
– votre souris,
– votre carte graphique,
– votre écran.

Se renseigner sur votre carte graphique

La première étape consiste à se renseigner sur votre carte graphique. Si vous connaissez le modèle exact de votre carte graphique, vous pouvez passer au paragraphe suivant. Sinon, il va falloir partir aux renseignements pour connaître le modèle exact de votre carte graphique. Vous avez quatre sources d'informations :
– les références écrites directement sur votre carte graphique, à l'intérieur du chassis de votre ordinateur, si vous pouvez l'ouvrir,
– la documentation ou la facture de votre ordinateur,

– la commande **lspci** sous Linux, qui liste les périphériques PCI et AGP. Vous devez avoir un paragraphe qui commence par `vga compatible controller`, et à la suite le nom du modèle de votre carte graphique.
– si vous avez un Windows installé sur votre ordinateur, regardez dans le *Gestionnaire de périphériques*, dans la section *Carte Graphique*, pour avoir le nom de votre carte.

Comprendre l'accélération 3D sous X

Trois cas se présentent :
– Votre carte graphique ne possède pas d'accélération 3D matérielle : vous n'êtes pas concerné par ce paragraphe !
– Vous avez une carte graphique possédant une puce de marque 3Dfx, Intel, Matrox ou SiS : les pilotes graphiques libres déjà installés prennent en charge l'accélération 3D, vous n'avez donc rien à faire !
– Vous avez une carte ATI ou nVidia. Vous avez alors le choix entre deux pilotes : un pilote libre inclus dans X.Org, et un pilote propriétaire. Les pilotes propriétaire fonctionnent en 2D comme en 3D. Pour les cartes nVidia, le pilote libre marche très bien, de façon plus fiable que le pilote propriétaire, mais ne tire pas parti de l'accélération matérielle 3D de la carte. Pour les cartes ATI, le pilote libre sait utiliser les capacités 3D de la plupart des cartes ATI.

Installer un pilote propriétaire

Si vous souhaitez utiliser le pilote propriétaire nVidia, installez le paquet *nvidia-glx*. Vous devez pour cela avoir activé les dépôts de paquets non libres (18) :

```
# aptitude install nvidia-glx
```

De même, pour les cartes ATI, il faut installer le paquet *fglrx-driver*.

Le fichier de configuration d'X.Org

La configuration du serveur X est contenue dans le fichier `/etc/X11/xorg.conf`. Par défaut, ce fichier est absent, et le serveur X construit automatiquement une configuration adaptée au matériel détecté. Vous pouvez cependant préparer votre propre configuration en vous basant sur mon fichier d'exemple :

```
# cp ~/fichiers-config/xorg.conf /etc/X11/
```

ou :

```
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
   xorg.conf
# cp xorg.conf /etc/X11/
```

Les sections

Le fichier est divisé en plusieurs sections qui ont des liens entre-elles :

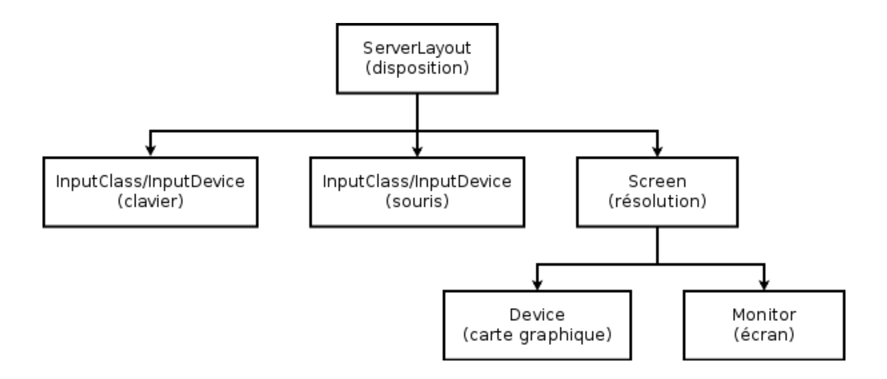


FIGURE 24.1 – Liens entre les sections

Les liens entre les sections sont assurés par des identifiants appelés *Identifier* : ainsi, la section **InputDevice** correspondant au clavier est référencée par son *Identifier* dans la section **ServerLayout**. En réalité, ces liens sont souvent omis, et déterminés automatiquement par le serveur X. Si vous modifiez un *Identifier* qui est référencé quelque part de façon explicite, n'oubliez pas de le modifier également à cet endroit.

Il existe aussi des sections indépendantes des autres :

- une section **Files** qui contient les chemins des polices de caractères,
- une section optionnelle **ServerFlags** qui peut contenir un certain nombre d'options,
- une section **Module** qui contient la liste des modules à charger,
- une section **DRI** qui contient les permissions pour l'utilisation du DRI (direct rendering infrastructure).

Nous vous précisons ci-dessous les paramètres importants dans chaque partie, en suivant l'ordre du fichier :

Le clavier

La configuration peut s'effectuer pour un clavier particulier dans une section **InputDevice**, ou pour tous les claviers dans une section **InputClass** :

- L'option **XkbModel** détermine le modèle physique du clavier (102 touches, 104 touches, 105 touches, Internet Logitec. . .).
- L'option **XkbLayout** permet de préciser la nationalité du clavier (français, américain, etc.).
- L'option **XkbVariant** permet de sélectionnerr une variante de la disposition de clavier (standard, dvorak, bépo. . .).

```
Section "InputClass"
        Identifier      "Claviers"
        MatchIsKeyboard "true"
        Option          "XkbRules"      "xorg"
        Option          "XkbModel"      "pc105"
        Option          "XkbLayout"     "fr"
  Option    "XkbVariant"  "oss"
EndSection
```

La souris

Comme pour les claviers, la configuration peut s'effectuer pour une souris particulière dans une section **InputDevice**, ou pour toutes les souris dans une section **InputClass** :

- L'option **Device** définit le périphérique correspondant à la souris.
- L'option **Protocol** définit le langage parlé par la souris.
- L'option **Emulate3Buttons** doit être activée si vous avez une souris 2 boutons et que vous voulez pouvoir faire du copier-coller sous X quand même (ce sera expliqué au Chapitre 26).

Cela donne par exemple :

```
Section "InputClass"
        Identifier      "Souris"
        MatchIsPointer  "true"
        # Décommentez la ligne suivante si vous avez une ↩
            souris 2 boutons
        #Option          "Emulate3Buttons"       "true"
EndSection
```

Et pour un pavé tactile :

```
Section "InputClass"
    Identifier "Pavés tactiles"
    MatchIsTouchpad     "true"
    Driver              "synaptics"
    Option              "TapButton1"    "1"
    Option              "TapButton2"    "2"
    Option              "TapButton3"    "3"
```

La carte graphique

- Le paramètre **Driver** fixe le pilote utilisé par le serveur X pour votre carte graphique. Le nom du pilote est normalement déterminé automatiquement par le serveur graphique au démarrage. Vous trouverez des informations intéressantes à ce sujet dans le répertoire `/usr/share/doc/xserver-xorg`.

```
Section "Device"
        Identifier      "Generic Video Card"
        Driver          "nvidia"
EndSection
```

L'écran

- Le paramètre facultatif **HorizSync** précise la plage des fréquences de synchronisation horizontale en kHz.
- Le paramètre facultatif **VertRefresh** précise la plage des fréquences de rafraîchissement verticale en Hz.
- L'option **DPMS** active l'option d'économie d'énergie si vous avez un écran qui supporte la norme DPMS.

Toutes ces informations techniques sont normalement écrites dans le manuel de l'écran. Il est normalement inutile de les préciser, mais cela peut être nécessaire si la meilleure résolution n'est pas utilisée spontanément

```
Section "Monitor"
        Identifier      "Generic Monitor"
        HorizSync       28-50
        VertRefresh     43-75
        Option          "DPMS"
EndSection
```

La résolution

– Le paramètre **Device** assure le lien avec la section concernant la carte graphique : il doit être exactement identique aux paramètres **Identifier** de la section *Device*.
– Le paramètre **Monitor** assure le lien avec la section concernant l'écran : il doit être exactement identique au paramètres **Identifier** de la section *Monitor*.
– Le paramètre **DefaultDepth** définit sur combien de bits sont codées les couleurs (1 bit = noir et blanc, 8 bits = 256 couleurs, 16 bits = 65536 couleurs, 24 bits = 16 millions). Elle renvoie à une des sous sections suivantes.

Il y a ensuite un certain nombre de sous-sections. Seule la section dont le paramètre **Depth** est égale au paramètre **DefaultDepth** est prise en compte par le serveur X.

Les paramètres des sous-sections sont les suivants :
– Le paramètre **Depth** définit sur combien de bits sont codées les couleurs.
– Le paramètre **Modes** définit une liste de résolutions de l'écran. Le Serveur X va choisir la plus haute résolution possible dans la liste.

```
Section "Screen"
        Identifier      "Default Screen"
        Device          "Generic Video Card"
        Monitor         "Generic Monitor"
        DefaultDepth    24
        SubSection "Display"
                Depth           8
                Modes           "1024x768" "800x600" "640x480 ↩
                    "
        EndSubSection
        SubSection "Display"
                Depth           16
                Modes           "1024x768" "800x600" "640x480 ↩
                    "
        EndSubSection
        SubSection "Display"
                Depth           24
                Modes           "1024x768" "800x600" "640x480 ↩
                    "
        EndSubSection
EndSection
```

La section ServerLayout

Elle définit l'agencement de votre bureau. Cela peut être utile si vous utilisez plusieurs écrans, claviers et souris... :
– Le paramètre **Screen** assure le lien avec la section concernant la résolution : il doit être exactement identique au paramètres **Identifier** de la section *Screen*.

– Le paramètre **InputDevice** assure le lien avec le clavier et la souris : il doit donc être présent deux fois.
– On peut éventuellement rajouter le paramètre **OffTime** qui fixe le nombre de minutes d'inactivité au bout duquel un écran DPMS se met en mode d'économie d'énergie. Pour que ça marche, il faut également avoir activé l'option *DPMS* dans la section *Monitor*.

Cela donne par exemple :

```
Section "ServerLayout"
        Identifier      "Default Layout"
        Screen          "Default Screen"
        InputDevice     "Generic Keyboard"
        InputDevice     "Configured Mouse"
        Option          "OffTime"           "20"
EndSection
```

La section DRI

Cette section permet de fixer les permissions pour l'utilisation du DRI. Si vous n'utilisez pas le DRI (parce que vous avez une carte nVidia par exemple), alors ne mettez pas cette section dans votre fichier de configuration. Avec la section d'exemple ci-dessous, vous donnez le droit d'utiliser DRI à tous les utilisateurs du système :

```
Section "DRI"
       Mode    0666
EndSection
```

Fin...

Une fois que le fichier de configuration est au point, vous n'avez plus qu'à enregistrer les changements et à lancer le serveur X :

```
% startx
```

ASTUCE

Si vous souhaitez faire des essais avec diverses configurations, vous pouvez toujours fermer le serveur X en utilisant le menu *Système* > *Fermer la session*.

Chapitre 25

Le bureau GNOME

GNOME est un grand projet ayant pour but de construire un environnement graphique libre et simple d'utilisation. Il est basé sur une bibliothèque graphique libre appelée GTK+.

25.1 Découverte de GNOME

L'interface de GNOME est assez classique et très simple à comprendre. Elle se compose de trois parties :

- Une barre supérieure, qui contient notamment les trois parties du menu GNOME. Dans *Applications*, vous trouverez tous les programmes installés sur votre système, rangés par catégories. Le menu *Raccourcis* vous donne accès aux répertoires couramment utilisés, et le menu *Bureau* vous permet de régler votre bureau et d'administrer votre système.
- Un bureau, avec des icônes correspondant aux répertoires les plus utilisés.
- Une barre inférieure, qui contient essentiellement la liste des fenêtres, et un sélecteur d'*espaces de travail* (nous verrons tout à l'heure ce que cela signifie).

Si l'apparence de votre bureau GNOME ne vous satisfait pas, vous pouvez déja modifier son aspect dans le menu *Bureau* > *Préférence* > *Thème*

25.2 Utiliser GNOME

Là encore, je ne vais pas faire de grands discours : promenez votre souris un peu partout et vous découvrirez par vous-même.



FIGURE 25.1 – Le logo GNOME

Utiliser le gestionnaire de fichiers Nautilus

Le gestionnaire de fichier de GNOME s'appelle Nautilus. Pour le lancer, ouvrez votre répertoire personnel.

Vous avez alors un gestionnaire de fichiers tout simple comme sur la Figure 25.2.

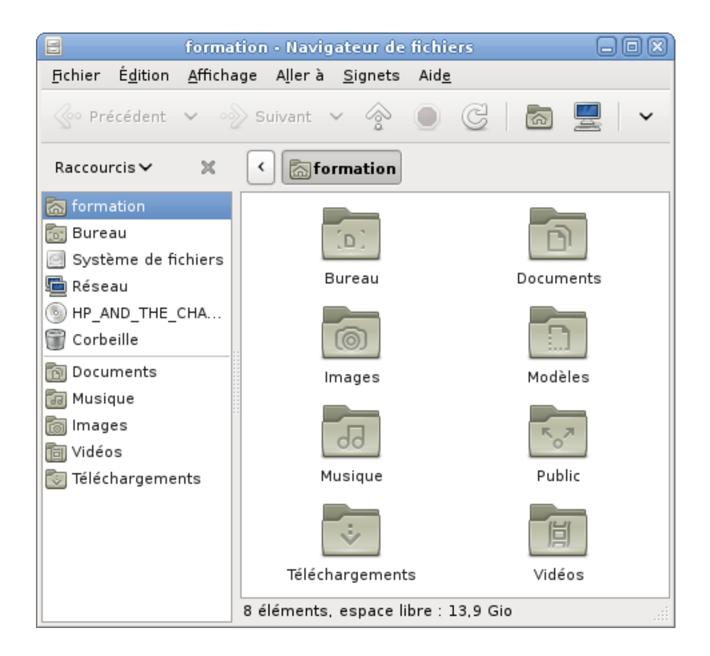


FIGURE 25.2 – Nautilus

Fermer le serveur graphique

Pour fermer GNOME et le serveur graphique, cliquez sur *Bureau*, puis sur *Clore la session*.

Chapitre 26

Les bases de Linux en mode graphique

Dans ce chapitre, vous allez apprendre à taper des commandes en mode graphique, à lancer des applications graphiques, à utiliser des bureaux virtuels, à faire du copier-coller sous X et à éditer des textes.

26.1 Un terminal sous X

GNOME propose un *émulateur de terminal*, qui simule un véritable terminal. Vous le trouverez dans *Applications* > *Accessoires* > *Terminal*.

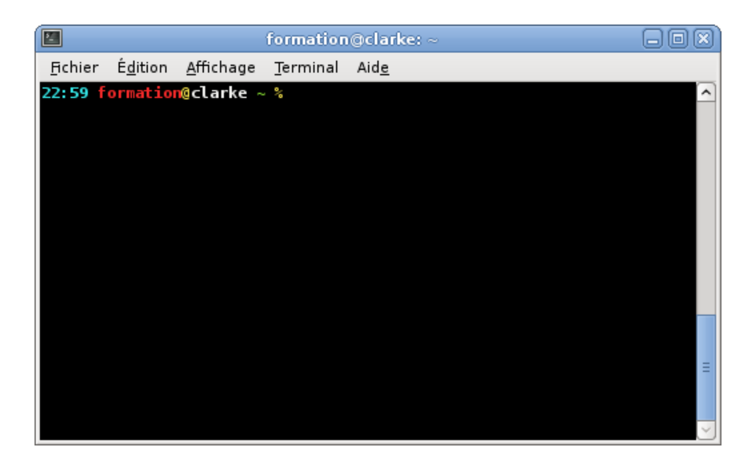


FIGURE 26.1 – Le terminal GNOME

Note

Outre celui de GNOME, il existe de nombreux émulateurs de terminaux : xterm, Aterm, Eterm, Konsole, rxvt-unicode, etc. Chaque émulateur de terminal propose un aspect et des fonctionnalités différents, par exemple des onglets ou de la transparence.

ASTUCE

Vous aurez probablement souvent besoin d'utiliser le terminal. En faisant glisser son icône du menu des Applications vers la barre supérieure, vous pouvez ajouter un petit raccourci.

Dans les préférences du bureau, vous pouvez également définir un raccourci clavier pour lancer un terminal d'une simple combinaison de touches.

Si vous utilisez Nautilus, le gestionnaire de fichiers de GNOME, vous pouvez également installer le paquet *nautilus-open-terminal* qui vous permettra d'ouvrir un émulateur de terminal dans le répertoire courant par un simple clic droit.

26.2 Faire du copier-coller sous X

Pour copier du texte, comme sous Windows, vous pouvez utiliser les commandes *Copier* et *Coller* ou leurs équivalents au clavier.

Le serveur graphique X.Org offre une possibilité bien plus efficace, le copier-coller *à la X-Window* :

1. sélectionnez du texte avec le bouton gauche de votre souris,
2. placez le curseur là où vous voulez coller le texte,
3. collez le texte en cliquant sur le bouton du milieu de votre souris ou, si votre souris n'a que deux boutons, en cliquant sur le bouton gauche et le bouton droit en même temps (il faut alors que vous ayez activé l'option **Emulate3Buttons** dans le fichier de configuration de votre serveur graphique).

26.3 Lancer et tuer une application graphique

Lancer une application graphique

Avec le menu Applications

Vous pouvez chercher l'application graphique dans les catégories du menu *Applications* (toutes les applications graphiques y sont ajoutées et rangées automatiquement lors de leur installation).

Avec le lanceur d'applications

Vous pouvez utiliser la combinaison de touches Alt-F2 pour afficher une fenêtre de lancement d'applications. Vous pouvez alors y saisir le nom de l'application à lancer.

Depuis un terminal

Vous pouvez également lancer une application graphique depuis un terminal : tapez-y la commande correspondante avec ses options suivie du caractère **&**, qui demande au programme de vous rendre la main sur le Shell. Par exemple, pour lancer le programme xclock :

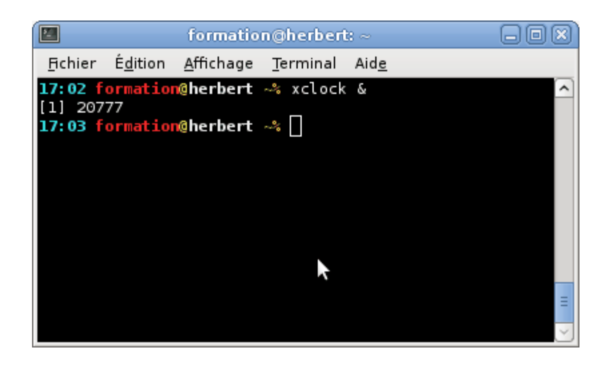


FIGURE 26.2 – Lancer depuis un terminal

Cette méthode permet de lancer facilement une application graphique en root : il suffit de devenir root dans le terminal avec la commande **su**, avant de lancer le programme.

Note

La quasi-totalité des applications graphiques se lancent avec la commande qui porte leur nom !

Tuer une application graphique

Lancez le programme **xkill** depuis un terminal. Vous avez alors un curseur de souris spécial avec lequel vous allez cliquer sur l'application graphique que vous voulez tuer. Bien sûr, cette technique ne doit être utilisée que pour une application qui ne veut pas se fermer normalement.

26.4 Utiliser les bureaux virtuels

Les quatre rectangles gris situés en bas à droite de votre écran et reproduits sur la Figure 26.3 représentent vos *espaces de travail* ou *bureaux virtuels*. Vous vous trouvez actuellement dans le premier bureau, qui apparaît coloré en bleu.

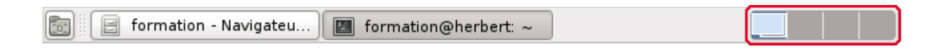


FIGURE 26.3 – Le sélecteur d'espaces de travail

Ces bureaux se comportent comme autant d'écrans distincts, ce qui vous permet d'organiser vos fenêtres. Quand vous lancez un logiciel, celui-ci affiche une fenêtre dans le bureau courant. Essayez maintenant de cliquer sur le rectangle du second bureau : cette fenêtre est restée sur le premier bureau ; elle n'est donc plus affichée ni listée dans la liste des fenêtres, mais n'est pas fermée pour autant.

Vous pouvez ainsi utiliser un bureau pour vos tâches administratives (déplacer, renommer des documents), un autre pour votre travail (bureautique), un pour naviguer

sur le Web et lire votre courrier, et un troisième pour écouter de la musique, sans risquer de vous perdre dans des dizaines de fenêtres.

ASTUCE

Si quatre espaces de travail ne vous suffisent pas, d'un clic droit sur la liste des bureaux virtuels, vous pouvez régler leur nombre.

Pour passer d'un bureau à un autre, vous pouvez cliquer sur les rectangles qui les représentent, ou utiliser les combinaisons Ctrl-Alt-gauche et Ctrl-Alt-droite. Ces combinaisons peuvent bien sûr être modifiées dans les préférences du bureau GNOME.

26.5 Vim sous X

Vous pouvez bien sûr utiliser Vim dans un *terminal*, mais vous pouvez aussi utiliser la version graphique de Vim :

```
# aptitude install vim-gnome
```

Il se lance avec la commande **gvim**.

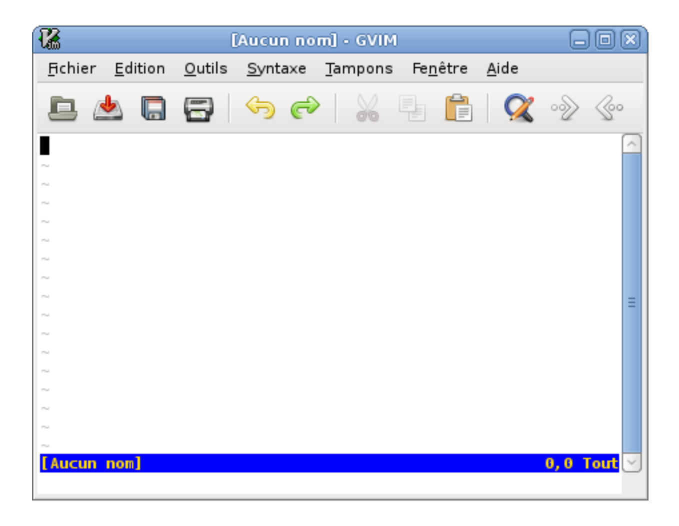


FIGURE 26.4 – Gvim

26.6 Vérifier que l'accélération matérielle 3D fonctionne

Si vous avez une carte graphique avec accélération matérielle 3D, vous pouvez vérifier que l'accélération matérielle fonctionne avec la commande **glxinfo** exécutée dans un terminal : les trois premières lignes doivent contenir :

```
name of display: :0.0
display: :0  screen: 0
direct rendering: Yes
```

S'il marque à la troisième ligne `direct rendering: No`, c'est que l'accélération matérielle 3D ne fonctionne pas.

Chapitre 27

Le Web, le courrier et les news en mode graphique

Nous arrivons enfin à la partie applicative de cette formation ! Le but de ce chapitre est d'installer deux logiciels que vous connaissez probablement déjà sous Windows : Firefox pour la navigation Web et Thunderbird pour le courrier et les news. Mozilla est un grand projet libre issu de la décision d'ouvrir le code source de Netscape, et il propose aujourd'hui ces deux logiciels non seulement pour GNU/Linux, mais aussi pour Windows et Mac OS X.

27.1 Installer Iceweasel

Debian n'a pas l'accord de la fondation Mozilla pour utiliser le nom « Firefox » ainsi que le logo associé. Sous Debian, ce navigateur porte donc le nom « Iceweasel », mais il est pour le reste parfaitement identique au produit d'origine.

Installer les paquets

Iceweasel est normalement déjà installé. En revanche, pour pouvoir l'utiliser avec une interface française (Figure 27.1), vous devez installer les fichiers de traduction :

```
# aptitude install iceweasel-l10n-fr
```



FIGURE 27.1 – Iceweasel

Surfer avec Iceweasel

Lancez Iceweasel à partir du menu des applications ou dans un terminal.

Si vous devez passer par un proxy pour aller sur le Web, allez dans le menu *Edition* > *Préférences* et entrez ses paramètres.

27.2 Le courrier et les news avec Icedove

Note

Le lecteur de courrier et de news proposé par GNOME s'appelle *Evolution*. Il est simple d'utilisation et parfaitement intégré au bureau. Pour cette formation, nous avons choisi de présenter le logiciel Icedove, qui est connu sous Windows sous le nom de Thunderbird, mais vous pouvez tout à fait essayer Evolution et choisir de l'utiliser si vous préférez.

Comme pour Firefox, Debian n'a pas l'accord de la fondation Mozilla pour utiliser le nom « Thunderbird ». Sous Debian, ce logiciel s'appelle donc « Icedove ». Ce logiciel est capable de relever des messages par POP et IMAP, de lire des forums de news et de trier votre courrier automatiquement.

Installer les paquets

```
# aptitude install icedove icedove-l10n-fr
```

Utiliser Icedove

Au premier lancement, un assistant se lance et vous propose de configurer un compte courrier ou news. Pour changer la configuration des comptes de courrier ou de news ou rajouter de nouveaux comptes, allez dans le menu *Edition* > *Paramètres des comptes* . La configuration du filtrage des messages se fait dans le menu *Outils* > *Filtres de messages*.

Lorsque vous le lancez pour la première fois, Icedove vous propose de configurer votre compte de courrier de façon assez automatisée (Figure 27.2). Par ailleurs, l'interface de la version GNU/Linux (Figure 27.3) est presque identique à celle de la version Windows (de Thunderbird) : je vous laisse donc découvrir ses capacités.

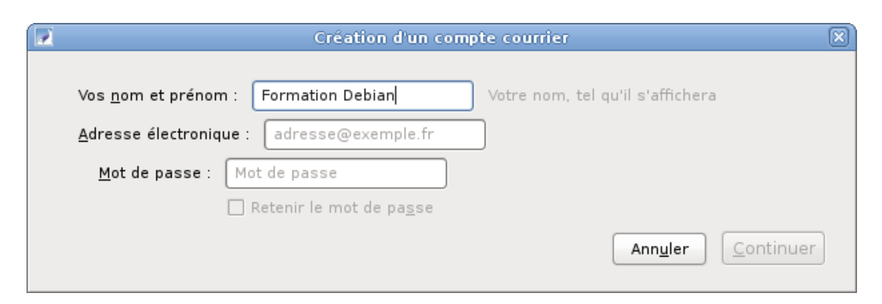


FIGURE 27.2 – Configuration d'un compte avec Icedove

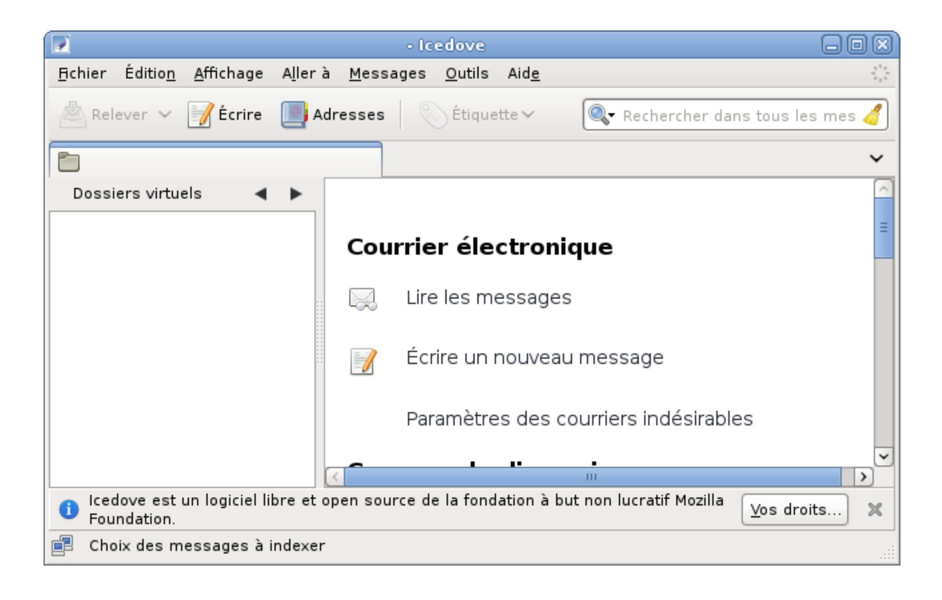


FIGURE 27.3 – Icedove

27.3 Un client FTP graphique

GNOME permet de se connecter à un serveur FTP, SSH, SMB ou WebDAV directement dans Nautilus. Une fois connecté à un serveur, vous pourrez même accéder aux

fichiers qui sont dessus dans vos applications, comme s'ils étaient sur votre ordinateur !

Pour cela, il faut aller dans le menu *Raccourcis > Se connecter à un serveur*. Puis choisir le type de serveur, par exemple *FTP public* ou *FTP (avec authentification)*, et régler les paramètres de connexion (Figure 27.4).

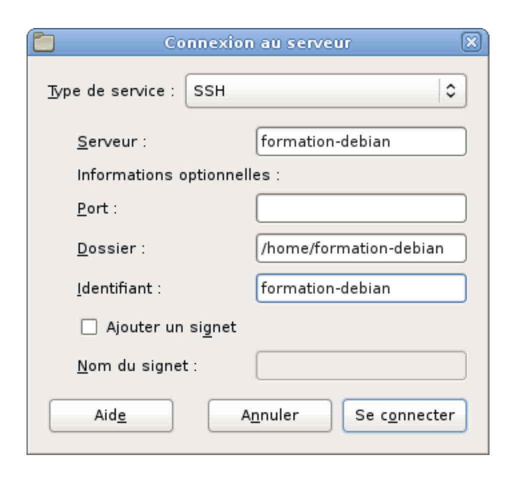


FIGURE 27.4 – Connexion à un serveur SSH

Si vous préférez utiliser un client FTP dédié à cela, vous pouvez installer gFTP, qui correspond au paquet *gftp-gtk*

Chapitre 28

La musique sous X

Nous vous proposons de découvrir un lecteur de musique de GNOME nommé Rhythmbox. Vous apprendrez également à encoder un CD audio en Ogg/Vorbis avec Sound Juicer.

28.1 Jouer de la musique avec Rhythmbox

Rhythmbox permet de lire les formats courants de musique, les CD audio, ainsi que de gérer une collection de musique et une liste de lecture. Ce lecteur de musique fait partie du bureau GNOME, et devrait donc déjà être installé.

Au premier lancement, vous arrivez dans l'interface de Rhythmbox, Figure 28.1

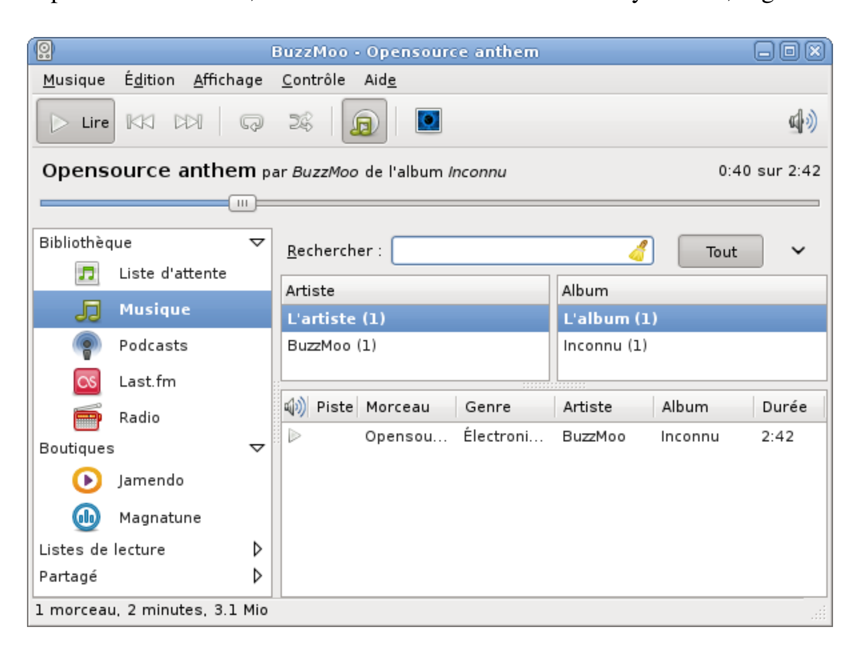


FIGURE 28.1 – Rhythmbox

Rhythmbox est capable de gérer une discothèque. Celle-ci est pour l'instant vide.

Vous pouvez la remplir en utilisant le menu *Musique* > *Importer un dossier* et en sélectionnant un répertoire contenant de la musique

Dans la partie gauche, vous pouvez afficher votre bibliotèque de musique ou la liste de lecture. Pour lire un morceau, double-cliquez dessus. Pour ajouter un morceau à la liste de lecture, faites-l'y simplement glisser, depuis la bibliothèque ou depuis un explorateur de fichiers Nautilus.

ASTUCE

Rhythmbox dispose de nombreux greffons qui permettent d'étendre ses capacités de lecteur de musique. Certains permettent d'utiliser des services externes comme Last.fm ou Jamendo, que je vous invite à découvrir.

28.2 Écouter ou encoder un CD en Ogg/Vorbis avec Sound Juicer

Pourquoi Ogg/Vorbis et pas MP3 ?

Le format MP3 a fait l'objet de dépôt de brevets, et son utilisation est donc soumise au bon vouloir de ses détenteurs. Le projet Xiph.org a donc développé un format de compression audio ouvert et libre de tout brevet, ainsi que les outils nécessaires à la compression et à la décompression. Aujourd'hui, le format Ogg/Vorbis est plus efficace en terme de qualité pour une même compression que le format MP3 et devient de plus en plus populaire.

Compresser un CD en Ogg/Vorbis

Mettez un CD audio dans le lecteur : Sound Juicer se lance automatiquement. Si vous avez une connexion Internet, il va consulter une base d'informations sur les CD pour y récupérer le nom de l'auteur, de l'album et des chansons (Figure 28.2).

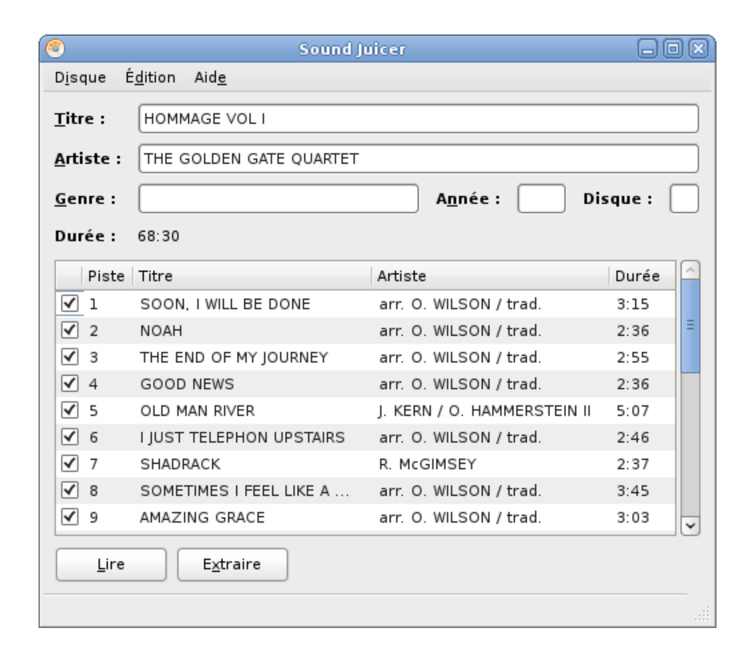


FIGURE 28.2 – Sound Juicer

Nous allons maintenant jeter un œil à la configuration. Allez dans *Édition* > *Préférences*.

Vous pouvez alors régler le répertoire dans lequel vous souhaitez stocker votre musique, l'arborescence qui doit être créée en fonction des albums que vous encodez, ainsi que le format de sortie. Vérifiez que vous encoderez bien en *Ogg multimédia*. Fermez ensuite la fenêtre de réglages.

Toutes les pistes du disque sont normalement cochées, il suffit de cliquer sur le bouton *Extraire* pour commencer l'encodage.

Vous préfèrez quand même utiliser le format MP3...

Les détenteurs des brevets relatifs au format MP3 exigent normalement l'acquisition d'une licence pour distribuer des logiciels d'encodage pour ce format, mais ont choisi de ne pas faire valoir leur droit pour les logiciels libres. Il existe donc des encodeurs MP3 libres, mais compte tenu de la fragilité de leur existence légale, ils ne sont pas distribués par Debian. Si vous voulez quand même encoder en MP3, vous pouvez ajouter à vos sources de paquets un dépôt Debian-multimédia, puis installer le paquet *gstreamer0.10-lame*, et enfin suivre les explications de GnunuX.

Chapitre 29

VLC media player, un lecteur multimédia

29.1 VideoLAN, c'est quoi ?

VideoLAN est un projet libre développé par des élèves de l'École Centrale Paris et des dizaines de développeurs à travers le monde. L'objectif est de diffuser de la vidéo numérique haute résolution sur un réseau informatique.

VLC media player, dont le nom vient de sa fonction d'origine, *VideoLAN Client*, est capable de lire :
– des fichiers MPEG-1, MPEG-2 et MPEG-4 / DivX, etc.
– des DVD et des VCD,
– depuis une carte satellite,
– depuis le réseau (utile pour la solution globale client/serveur).

Pour plus d'informations, je vous invite à visiter le site web de VideoLAN et en particulier la page des fonctionnalités de VLC media player.

29.2 Installer VLC

AVERTISSEMENT

La plupart des DVD commerciaux sont chiffrés pour rendre leur lecture impossible sans acheter un décodeur approuvé, dans le but de limiter les possibilités de copie. Heureusement, la bibliothèque libre *libdvdcss* a été écrite pour briser cette restriction. Il est donc *techniquement possible* de lire des DVD sous GNU/Linux sans acheter de lecteur commercial.

En revanche, la loi DADVSI *interdit* le contournement de *mesures techniques de privation*.

Le Conseil d'État a toutefois estimé que les mesures techniques de privation ne pouvaient s'opposer à l'interopérabilité. Vous devriez donc normalement pouvoir lire vos DVD sous GNU/Linux sans craindre de poursuites, mais il est préférable d'éviter d'acheter des œuvres utilisant de telles mesures de privation.

La *libdvdcss* étant déja hors la loi aux État-Unis, Debian ne propose donc pas de

paquet pour cette bibliothèque. Si vous souhaitez pouvoir lire des DVD chiffrés, vous allez donc devoir utiliser un dépôt non officiel, en ajoutant la ligne suivante au fichier /etc/apt/sources.list :

```
deb http://unofficial.debian-maintainers.org/ squeeze main
```

Installez ensuite VLC, ainsi que la *libdvdcss* si vous souhaitez pouvoir lire des DVD protégés contre la lecture :

```
# aptitude update
# aptitude install libdvdcss2 vlc
```

Note

VLC dispose de plusieurs interfaces, et peut ainsi être contrôlé de différentes manières, y compris depuis un navigateur Web, une console ou une connexion réseau. Debian propose par défaut l'interface graphique *wxWidgets* représentée sur la Figure 29.1.

29.3 Lancer VLC

Pour lancer VLC, il suffit d'utiliser la commande **vlc**. Ensuite, l'interface est intuitive et tout se fait au clic !

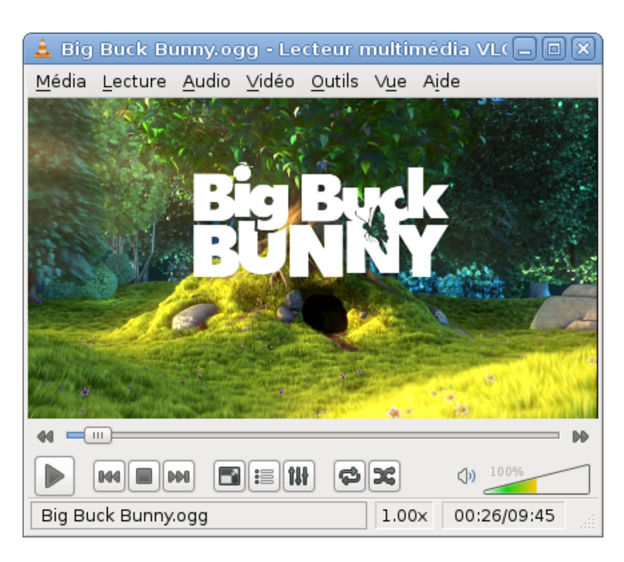


FIGURE 29.1 – Interface wxWidgets de VLC

Chapitre 30

La bureautique avec OpenOffice.org

30.1 OpenOffice.org

OpenOffice.org (souvent abrégé en « OOo ») est une suite bureautique libre très complète, qui comprend un traitement de texte, un tableur, un outil pour faire des présentations et un logiciel de dessin. Elle a l'avantage d'être également disponible sous Windows et Mac OS X, tout en étant compatible avec les formats de fichiers de Microsoft Office : vous pouvez ouvrir des documents Word, Excel et PowerPoint, et enregistrer vos documents dans ces différents formats.

OpenOffice.org est le descendant de StarOffice 5.2 qui était une suite bureautique gratuite éditée par Sun Microsystems. Sun a décidé d'ouvrir le code source de StarOffice et de lancer le projet OpenOffice.org, mais Sun continue de sortir de nouvelles versions de StarOffice désormais payantes et comportant des fonctions supplémentaires.

30.2 Installer OpenOffice.org

OpenOffice.org est normalement déjà installé sur votre système Debian. En revanche, c'est à vous d'installer sa traduction française, ainsi que les outils francophones comme le correcteur d'ortographe :

```
# aptitude install openoffice.org-l10n-fr openoffice.org-help ↩
    -fr myspell-fr-gut
```

30.3 Lancer OpenOffice.org

Les différents composants d'OpenOffice.org sont accessibles par le menu *Applications > Bureautique* :

Fonction	**composant d'OOo**	**équivalent Microsoft Office**
Traitement de texte	Writer	Word
Tableur	Calc	Excel

Fonction	**composant d'OOo**	**équivalent Microsoft Office**
Présentation	Impress	PowerPoint
Dessins, schémas	Draw	inexistant
Bases de données	Base	Access

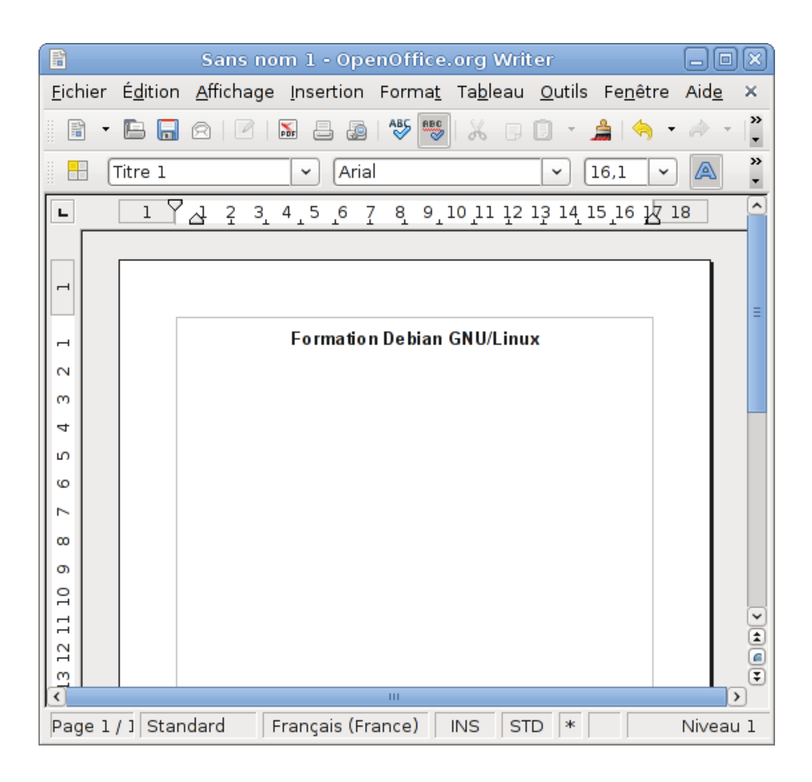


FIGURE 30.1 – Writer, le traitement de texte d'OpenOffice.org

Pour l'utilisation proprement dite, nous vous laissons découvrir : c'est très simple, et ça ressemble beaucoup à Microsoft Office ! Si vous préférez être guidé dans votre apprentissage, il existe des guides et des didacticiels en français disponibles sur la partie francophone du site officiel d'OpenOffice.org.

ASTUCE

OpenOffice.org est capable d'exporter ses documents au format PDF. C'est très pratique pour diffuser les documents crées avec OpenOffice.org à des gens qui ne se sont pas encore convertis !

Chapitre 31

La manipulation d'images

31.1 Afficher des images

Pour afficher une image, double-cliquez simplement dessus : elle s'ouvre alors avec Eye of GNOME, l'œil de GNOME (Figure 31.1).

Note

Vous pouvez parcourir les images d'un répertoire en utilisant les flèches, ou afficher une série d'images sous forme de diaporama, en utilisant le menu *Affichage > Diaporama (**F5**)*.

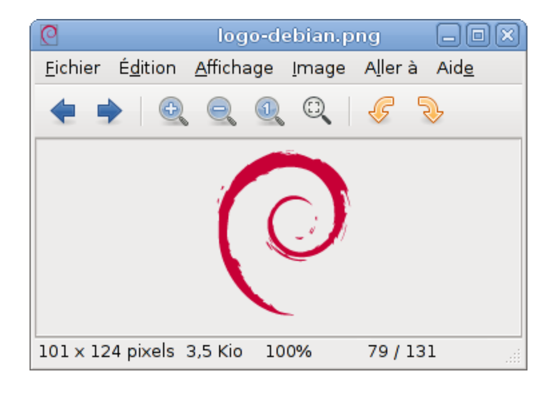


FIGURE 31.1 – Eye of GNOME

31.2 Traiter des images en ligne de commande avec ImageMagick

```
# aptitude install imagemagick
```

ImageMagick est un ensemble d'outils en ligne de commande très pratiques :

- La commande **display** permet de visualiser une image :

```
% display nom_du_fichier_image &
```

– La commande **identify** permet d'avoir des informations sur l'image :

```
% identify mon_image.jpg
mon_image.jpg JPEG 861x306 DirectClass 8-bit 142kb 0.0u ↩
    0:01
```

– La commande **convert** permet de convertir d'un format à un autre ou de transformer une image. Par exemple, pour passer une image du format JPEG au format EPS :

```
% convert image1.jpg image1.eps
```

On peut aussi redimensionner une image (ici, on la redimentionne sans modification de rapport, de façon à ce qu'elle rentre dans un carré de 100×100) :

```
% convert -geometry 100x100 image_originale.jpg ↩
    image_redimensionnée.jpg
```

– La commande **mogrify** permet d'effectuer des transformations, tout comme **convert**, mais en traitant les images sur place, sans créer de nouveau fichier. Cela s'avère très utile pour redimentionner toute une série de photos. Sa syntaxe est identique à celle de **convert**, avec une seule image. Si vous précisez plusieurs images, elles seront toutes modifiées.

31.3 La retouche d'images avec GIMP

GIMP est un logiciel libre de retouche d'image, souvent comparé à Photoshop. Il est normalement déjà installé sur votre système, mais si ce n'était pas le cas :

```
# aptitude install gimp
```

L'interface de GIMP se décompose en trois fenêtres, dont deux sont reproduites en Figure 31.2. La première donne accès à la boîte à outils de base. Une seconde fenêtre contient l'image en cours de traitement, ainsi qu'une barre qui donne accès aux manipulations sous forme de menus. Enfin, une troisième fenêtre contient les listes des calques, canaux et chemins de l'image.

GIMP est assez simple à utiliser si vous avez déja l'habitude d'utiliser ce genre de logiciel. Si vous découvrez le traitement d'images, je vous conseille de suivre quelques tutoriels de gimp-fr.

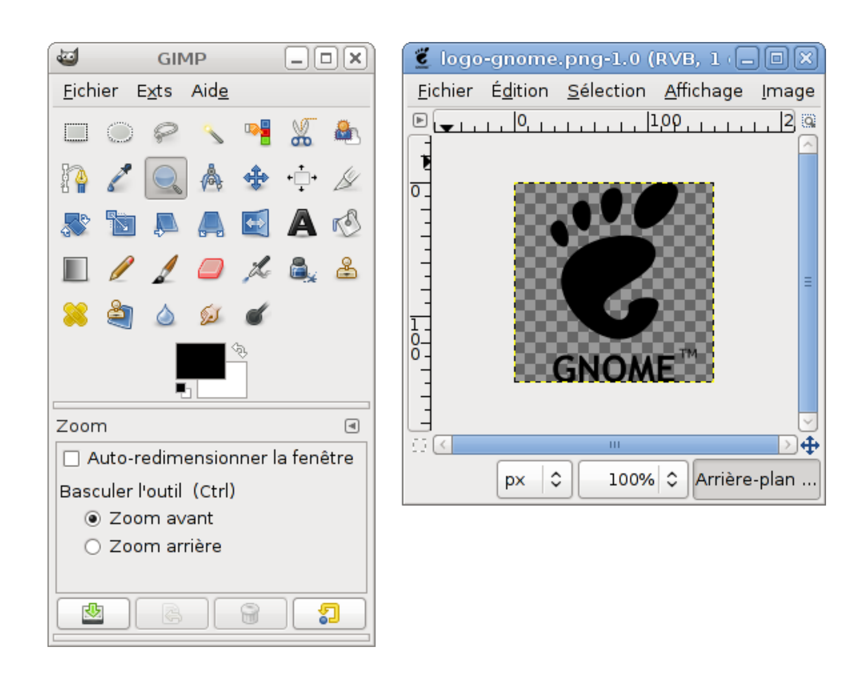


FIGURE 31.2 – GIMP, fenêtres d'outils et d'image

Note

GIMP permet aussi de faire facilement des captures d'écran sous X : menu *Fichier*, *Acquisition*, *Capture d'écran*.

Chapitre 32

La messagerie instantanée avec Pidgin

Pidgin, anciennement nommé Gaim, est un client de messagerie instantanée capable de se connecter à bon nombre de protocoles de messagerie : Jabber, AIM, ICQ, IRC, MSN...

Note

Jabber, alias XMPP est le réseaux de messagerie instantanée d'Internet, qui utilise le protocole standard XMPP. Contrairement aux autres réseau, Jabber est ouvert et normalisé. Il permet en outre une interconnexion avec les autres protocoles.

Pidgin est un bon logiciel de messagerie instantanée multi-protocole, mais si vous comptez n'utiliser que Jabber, je vous conseille de vous tourner vers un logiciel dédié, comme par exemple Gajim.

32.1 Installation de Pidgin

Si vous suivez notre formation depuis le début, Pidgin est normalement déjà installé sur votre système. Sinon, Installez simplement le paquet *pidgin* :

```
# aptitude install pidgin
```

32.2 Configurer un compte

Vous pouvez lancer Pidgin depuis le menu *Applications* > *Internet*. Pidgin peut gérer plusieurs comptes chez différents fournisseurs de messagerie instantanée : comme vous n'en avez pour le moment défini aucun, une fenêtre vous propose de le faire maintenant. Cliquez donc sur le bouton *Ajouter*.

Note

Vous pouvez ouvrir un compe Jabber depuis Pidgin. En revanche, pour utiliser un les protocoles propriétaires comme MSN ou AIM, vous devez déja posséder un compte sur ces réseaux.

Réglez les paramètres du compte (Figure 32.1) : l'alias local est le nom qui désignera votre compte dans la liste des comptes. Si vous souhaitez créer un compte Jabber/XMPP, cochez l'option *Créer ce nouveau compte sur le serveur*. Validez en cliquant sur *Enregistrer*.

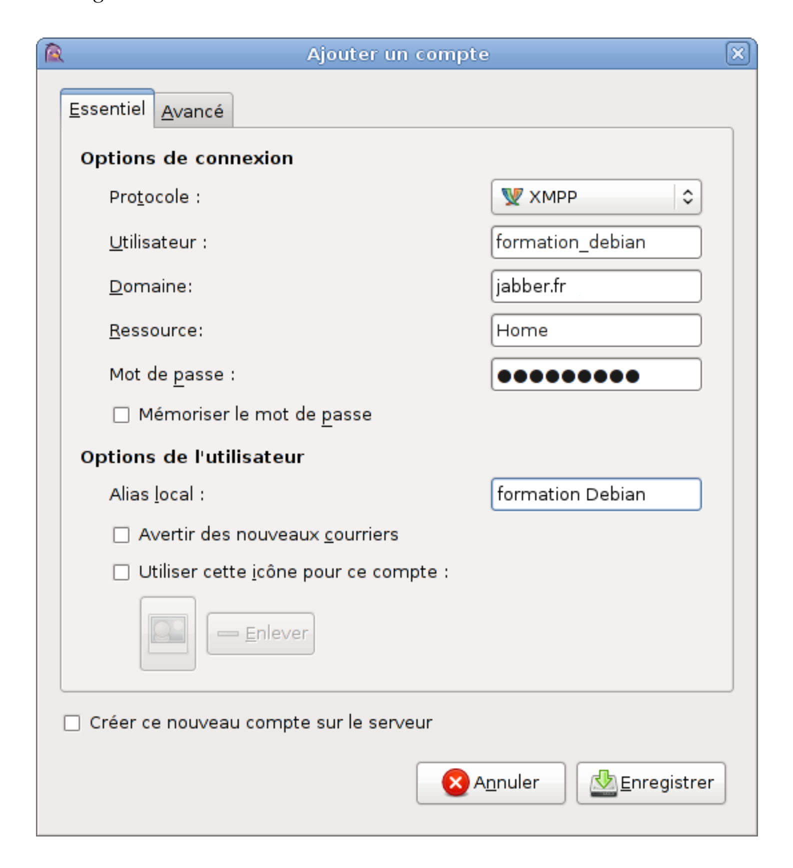


FIGURE 32.1 – Ajout d'un compte

32.3 Utiliser Pidgin

La fenêtre principale de Pidgin est la liste des contacts, reproduite en Figure 32.2, que vous pouvez appeler à tout moment en cliquant sur l'icône représentant une bulle de bande dessinée dans la zone de notification. À partir de là, vous pouvez ajouter des contats, ouvrir des fenêtres de discussion, ou encore accéder à la liste des comptes.

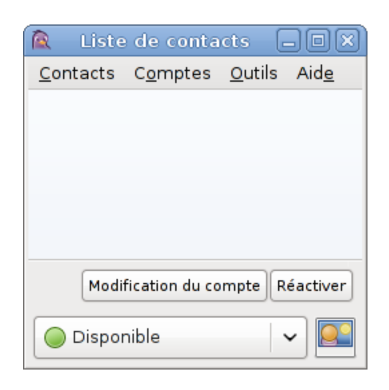


FIGURE 32.2 – Pidgin : la liste des contacts

Note

Les possibilités de Pidgin sont nombreuses : vous pouvez également gérer des groupes ou des salons de discussion. . .

Chapitre 33

La téléphonie sur IP avec Ekiga

Ekiga est un logiciel de téléphonie sur IP, qui permet de se téléphoner ou de faire de la vidéo conférence via Internet à deux ou à plusieurs à condition de disposer d'une bonne connexion à Internet de type ADSL. Ils utilise les normes SIP et H.323, ce qui permet d'utiliser Ekiga alors que la personne distante utilise par exemple Microsoft NetMeeting !

33.1 Installer et configurer Ekiga

Note

Je n'ai personnellement utilisé Ekiga qu'avec un micro, sans webcam. Si vous voulez faire de la vidéo conférence, il faut avoir une webcam prise en charge par le pilote video4linux du noyau.

Installez le paquet de Ekiga :

```
# aptitude install ekiga
```

Lancez Ekiga avec la commande du même nom ou depuis le menu de GNOME.



FIGURE 33.1 – Ekiga

La première fois que vous lancez Ekiga, un assistant vous permet de régler les paramètres les plus importants. Si vous n'avez pas encore de compte de téléphonie, vous aurez alors la possibilité d'en créer un.

Si vous disposez déjà d'un compte SIP (les meilleurs FAI en proposent un avec leurs offres groupées), vous pouvez configurer Ekiga pour l'utiliser, par le menu *Édition > Comptes*.

33.2 Passer un appel

C'est très simple : tapez l'adresse SIP de votre interlocuteur dans la case du haut après *sip:* et cliquez sur la prise (ou appuyez sur **Entrée**.

Si quelqu'un vous appelle, une fenêtre vous présentera le nom de la personne qui vous appelle et qui vous proposera de décrocher.

Dès que votre interlocuteur décroche, la conversation peut commencer. Ajustez les volumes du mixer et de vos hauts-parleurs pour bien entendre votre interlocuteur.

Si vous êtes gênés par la détection automatique du silence (qui permet de n'envoyer aucune données quand la personne ne parle pas), cliquez sur le petit bouton représentant l'écran d'un oscilloscope.

Chapitre 34

Graver des disques

34.1 Installer les logiciels de gravure

Le bureau GNOME propose plusieurs logiciels permettant de graver des CD. Nous vous proposons de d'utiliser pour cela Nautilus, l'explorateur de fichiers, et Brasero.

```
# aptitude install brasero
```

34.2 Effacer un disque réinscriptible

Il est possible d'effacer un disque avec Brasero, mais cela ne sert pas à grand chose : tentez simplement de graver sur un disque déja utilisé en suivant les explications suivantes, et, au moment de lancer l'écriture, il vous sera proposé d'effacer votre disque.

34.3 Graver un disque de données

Introduisez un disque dans votre lecteur. Si ce disque est vierge, vous pouvez double-cliquer sur l'icône qui apparaît alors sur le bureau. S'il est déja utilisé et que vous souhaitez remplacer son contenu, utilisez le menu *Raccourcis* > *Créateur de* CD/DVD.

Vous obtenez alors une fenêtre semblable à celle de la Figure 34.1, dans laquelle vous pouvez placer les documents que vous souhaitez. Pour cela, vous pouvez glisser, copier, couper ou coller des fichiers et des répertoires depuis un explorateur de fichiers.

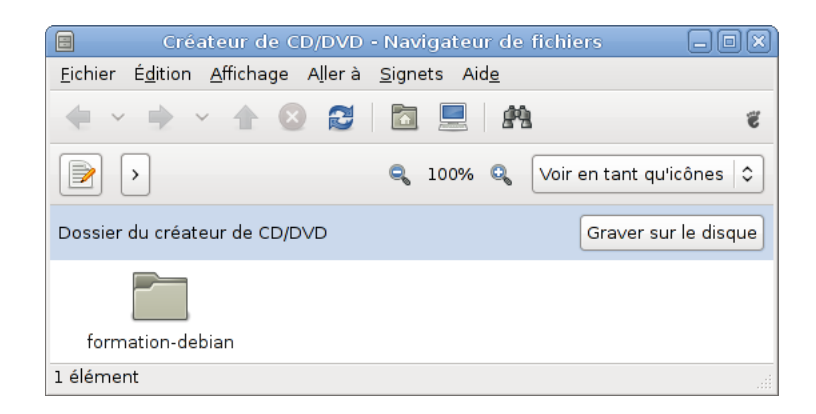


FIGURE 34.1 – Création d'un disque de données

34.4 Graver une image de disque

Si vous avez récupéré une image de disque, pour la graver, depuis un explorateur de fichiers Nautilus, il suffit de cliquer dessus du bouton droit, et d'utiliser l'entrée *Graver un disque*.

34.5 Graver un CD de musiques

Lancez Brasero depuis le menu *Applications* > *Son et vidéo* > *Gravure de disque Brasero*. Choisissez ensuite de commencer un *Projet audio*. Brasero vous explique alors comment ajouter des pistes musicales : vous pouvez faire glisser des fichiers depuis un explorateur, faire des copier-coller ou encore utiliser l'explorateur intégré à la fenêtre de Brasero.

Chapitre 35

Jouons un peu

On reproche souvent aux systèmes GNU/Linux de ne pas disposer de jeux vidéo. Certes, la plupart des jeux commerciaux ne sont disponibles que pour Windows, et parfois MacOS, mais il existe une multitude de jeux libres, dont certains sont excellents. Un bon nombre d'entre eux sont directement inspirés de jeux commerciaux, dont ils dépassent souvent la qualité. Il n'y a en fait que très peu de jeux originaux : après tout, Age of Empire n'est-il pas lui-même un clone de Warcraft ?

Il serait donc dommage de terminer cette partie de la formation sans vous présenter un florilège de jeux vidéo libres, et en profiter pour nous détendre un peu avant d'aborder les parties suivantes.

AVERTISSEMENT

Parmi les jeux que nous allons présenter, certains sont extrêmement addictifs. Si vous aimez finir à tout prix tous les niveaux des jeux que vous découvrer, nous vous invitons donc à sauter les sections 35.1, 35.2 et dans une moindre mesure 35.5, pour y revenir après avoir fini notre formation.

35.1 Frozen Bubble

Frozen Bubble est un jeu d'adresse, clone de Puzzle Bobble. Depuis 2002, ce jeu est devenu un culte sous GNU/Linux.



FIGURE 35.1 – Frozen Bubble à deux joueurs

Ce jeu correspond au paquet Debian *frozen-bubble*. Après l'avoir installé, vous pouvez le lancer depuis une console, ou en utilisant le menu de GNOME :

```
# aptitude install frozen-bubble
```

Frozen Bubble peut être joué seul, à deux sur un clavier (Figure 35.1, ou à plusieurs en réseau. Avec les flèches du clavier, vous contrôlez un canon qui permet de lancer des bulles vers le plafond. Le but est de former des blocs de plus de trois bulles de même couleur, afin de les faire tomber et de dégager le niveau.

35.2 X-Moto

X-Moto est un jeu d'adresse, clone d'Elasto Mania, dans lequel vous pilotez en 2D une moto extrêmement réactive dans un niveau de plate-formes, comme sur la Figure 35.2. Pour finir un niveau, vous devez passer par des points de contrôle, représentés par des fraises, puis atteindre une marguerite qui marque l'arrivée.



FIGURE 35.2 – Un niveau d'X-Moto

Pour jouer à X-Moto, installez simplement le paquet *xmoto*, puis lancez le jeu depuis le menu *Applications* ou depuis une console :

```
# aptitude install xmoto
```

Note

Depuis sa version 0.4, X-Moto tire parti de l'Internet et permet de récupérer des niveaux additionnels créés par d'autres joueurs et de soumettre ses scores au classement mondial. Sur le site officiel, il est également possible de définir des équipes.

35.3 La bataille pour Wesnoth

La bataille pour Wesnoth, encore appelé *The battle for Wesnoth* ou tout simplement *Wesnoth*, est un jeu de stratégie tour par tour. Fortement inspiré du jeu Heroes of Might and Magic, il se déroule dans un univers médiéval fantastique, dans lequel s'affrontent des morts-vivants, des elfes, des nains, des humains, des orcs, et des dracans (Figure 35.3).



FIGURE 35.3 – Un combat entre soldats et voleurs dans le monde de Wesnoth

Ce jeu est décomposé en plusieurs paquets, qui permettent d'installer indépendamment, le jeu seul, sa musique et ses nombreuses campagnes, ou encore son éditeur de niveaux. Si vous voulez disposer de l'ensemble du jeu, vous pouvez installer le paquet *wesnoth-all*. Dans tous les cas, je vous conseille d'installer la musique et la campagne principale, « l'héritier du trône » (*Heir to the throne*) :

```
# aptitude install wesnoth wesnoth-music wesnoth-httt
```

ASTUCE

Pour débuter, le jeu comporte une campagne d'apprentissage, disponible dans le menu d'accueil. Je vous conseille ensuite de jouer la campagne « l'histoire des deux frères » (*a tale of two brothers*), disponible dans le paquet *wesnoth-ttb*.

35.4 Wormux

Wormux se joue à deux ou plus, et met en scène des équipes formées par des mascottes de logiciels libres, le but étant d'exterminer les équipes adverses à l'aide d'armes plus ou moins fantaistes : lance-roquettes, grenades, fusil (Figure 35.4), lance-flammes, mais également super Tux, lance-GNU, grenade disco et gravité lunaire… Le nom de ce jeu laisse aisément deviner sa source d'inspiration, mais il est néanmoins garanti *sans ver*.



FIGURE 35.4 – Un combat amical entre GNUs et Tux

Pour jouer à Wormux, installer simplement le paquet *wormux*, puis lancez le jeu depuis le menu du bureau ou une console

35.5 Enigma

Enigma est à la fois un jeu de réflexion et d'adresse, dans lequel vous contrôlez à la souris une bille, à travers un labyrinthe parsemé de pièges et d'enigmes logiques. Ce jeu reprend des principes d'*Oxyd* et des jeux de pousse-pousse, et est livré avec une multitude de niveaux.

Ce jeu correspond au paquet *enigma*. Il est d'une telle richesse qu'il serait difficile d'en expliquer les principes : après avoir découvert des élastiques, des ressorts et des lasers, vous serez amenés à utiliser des automates à jetons, des tuyaux (Figure 35.5) et des extincteurs... Je vous conseille d'essayer directement les premiers niveaux, que l'on comprend assez rapidement, et de consulter la documentation officielle en cas de blocage.

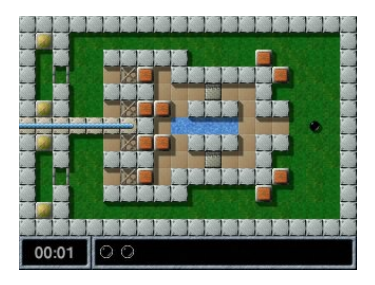


FIGURE 35.5 – Un niveau du jeu Enigma

35.6 Et bien d'autres

Il existe encore de très nombreux jeux libres. Si cela vous intéresse, je vous recommande particulièrement :

Pingus un jeu de réflexion semblable à Lemmings, dans lequel vous devez guider des manchots vers leur destin ;

SuperTux un jeu de plate-forme bien classique, à la Mario ;

SuperTuxKart le seul jeu de kart agressif pour PC, à ma connaissance ;

Chromium B.S.U. un *shoot'em up* aux graphismes magnifiques, dans lequel il faut veiller à ne laisser échapper personne

Enfin, vous pourrez découvrir d'avantage de jeux libres sur le site Jeux Libres.

Chapitre 36

Avant d'aller plus loin : un point sur la méthode

Vous arrivez à la fin des trois premières parties de cette formation. L'objectif de ces trois premières parties était de passer en douceur de Windows à GNU/Linux... j'espère que vous êtes maintenant capable de faire sous Debian la plupart des choses que vous faisiez avant sous Windows !

Avant d'aller plus loin et d'aborder des sujets plus avancés, il est bon de faire le point sur certaines méthodes utiles dans l'utilisation de Debian au quotidien. Vous connaissez normalement la plupart des commandes utilisées dans ce chapitre… mais pas forcément les méthodes exposées ci-dessous !

36.1 Méthode pour installer un logiciel

Le syndrome

« Mon ami m'a parlé d'un super navigateur pour GNU/Linux diffusé en logiciel libre, mieux que Mozilla… il s'appelle Galeon ! »

Votre réflexe d'ancien utilisateur de Windows sera peut-être de faire une recherche sur Google pour trouver le site Web du logiciel, de le télécharger et d'essayer de l'installer…

Mais comme c'est un logiciel libre pour GNU/Linux, alors il est probablement disponible (on dit aussi empaqueté ou *packagé*) pour Debian !

Faire une recherche dans la base des paquets Debian

Votre premier réflexe de nouvel utilisateur de Debian doit donc être de faire une recherche dans la base des paquets :

```
% apt-cache search galeon

galeon - GNOME web browser for advanced users
galeon-common - GNOME web browser for advanced users
```

Les résultats ci-dessus ont l'air de correspondre à ce que l'on cherche ! Le réflexe suivant est d'afficher les caractéristiques et la description de ce paquet :

```
% aptitude show galeon

Paquet~: galeon
État: non installé
Version~: 2.0.6-2
Priorité~: optionnel
Section~: gnome
Responsable~: Loic Minier <lool@dooz.org>
Taille décompressée~: 1778k
Dépend: gconf2 (>= 2.12.1-1), libbonobo2-0 (>= 2.15.0),  ↩
    libbonoboui2-0 (>= 2.15.1), libc6
         (>= 2.7-1), libgcc1 (>= 1:4.1.1), libgconf2-4 (>=  ↩
             2.13.5), libglade2-0 (>=
         1:2.6.1), libglib2.0-0 (>= 2.16.0), libgnome-desktop ↩
             -2 (>= 2.22.0), libgnome2-0
         (>= 2.17.3), libgnomeui-0 (>= 2.17.1), libgnomevfs2 ↩
             -0 (>= 1:2.17.90), libgtk2.0-0
         (>= 2.12.0), libnspr4-0d (>= 1.8.0.10), liborbit2  ↩
             (>= 1:2.14.10), libpango1.0-0
         (>= 1.20.3), libpopt0 (>= 1.14), libstdc++6 (>=  ↩
             4.1.1), libx11-6, libxml2 (>=
         2.6.27), procps, galeon-common, xulrunner (>= 1.9~)  ↩
             | xulrunner-1.9
Recommande: gnome-icon-theme (>= 1.1.3-1), yelp, scrollkeeper ↩
    , iso-codes,
            gnome-control-center
Suggère: mozplugger
Remplace: galeon-common
Fournit: gnome-www-browser, www-browser
Description~: Navigateur web de GNOME pour utilisateurs  ↩
    avancés
 Navigateur internet qui supporte les standards et s'intègre  ↩
     bien avec l'environnement de
 bureau GNOME. Il n'inclut pas de client de courriel, robot  ↩
     IRC, créateur de site web,
 etc. et utilise donc modérément les ressources. Le programme ↩
      utilise le moteur de rendu
 Gecko de Mozilla pour afficher les pages web. Il est donc  ↩
     pleinement fonctionnel et
 compatible avec les standards en plus d'afficher les pages  ↩
     rapidement.

 Page d'accueil~: http://galeon.sourceforge.net/
Site~:~http://galeon.sourceforge.net/

Étiquettes: filetransfer::ftp, filetransfer::http,  ↩
    implemented-in::c, interface::x11,
            network::client, protocol::ftp, protocol::http,  ↩
                protocol::ipv6, protocol::ssl,
            role::program, suite::gnome, uitoolkit::gtk, use ↩
                ::browsing, use::viewing,
```

```
            web::browser, works-with::text, works-with-format ↩
                ::html, x11::application
```

La description confirme que c'est bien le paquet qu'il nous faut. Les champs *Recommande* et *Suggère* conseillent d'installer également plusieurs autres paquets, par exemple *mozplugger*. La description de ce dernier nous apprend qu'il s'agit d'un plugin permettant d'utiliser plusieurs logiciels externes pour afficher des documents dans un navigateur : Galeon peut en tirer parti.

Installer le paquet

Pour installez Galeon, il suffit donc de lancer la commande suivante :

```
# aptitude install galeon
```

36.2 Méthode pour apprendre à se servir d'un logiciel

Pour les applications graphiques, il n'y a généralement pas de mystère : il suffit de lancer l'application avec le menu ou la commande ayant le même nom que l'application, et les fonctions devraient être toutes accessibles par l'interface.

Il en va tout autrement pour les applications en console et les serveurs. La méthode exposée ci-dessous va vous apprendre quelques réflexes à avoir pour apprendre à se servir d'un logiciel non graphique dont vous venez d'installer le paquet.

Ausculter le paquet

Tout d'abord, commencez par regarder la liste des fichiers qui étaient contenus dans le paquet :

```
% dpkg -L nom_du_paquet
```

Si les fichiers installés sont nombreux, vous pouvez n'afficher que ceux qui vous intéressent :

– la liste des exécutables :

  ```
  % dpkg -L nom_du_paquet | grep bin
  ```

– la liste des fichiers de configuration :

  ```
  % dpkg -L nom_du_paquet | grep etc
  ```

– la liste des fichiers contenant de la documentation :

  ```
  % dpkg -L nom_du_paquet | grep doc
  ```

– la liste des manuels :

  ```
  % dpkg -L nom_du_paquet | grep man
  ```

Lire la documentation et les manuels

Maintenant que vous cernez mieux le contenu du paquet, il est fortement conseillé d'explorer le répertoire `/usr/share/doc/nom_du_paquet/`. Ce répertoire doit normalement contenir :

- un fichier `README.Debian`, qui contient des infos sur la façon dont a été fait le paquet (sa lecture est fortement conseillée, car il contient des explications sur les différences éventuelles entre le logiciel tel qu'il est disponible sur Internet et le logiciel tel qu'il est packagé dans la Debian) ;
- un fichier `README`, qui est le README du logiciel ;
- un fichier `changelog.Debian`, qui contient l'historique du paquet Debian ;
- un fichier `changelog`, qui contient l'historique du logiciel ;
- un fichier `copyright`, qui contient le texte de la licence du logiciel ;
- éventuellement d'autres fichiers contenant de la documentation sur le logiciel au format texte ou HTML.

Note

Certains fichiers sont compressés (extension `.gz`) ; pour les lire, utilisez **zless**, **most** ou **vim**, qui sont capables de faire la décompression à la volée.

Attention

Pour les logiciels qui ont une documentation volumineuse, ce qui est souvent le cas des logiciels serveurs, la documentation est parfois contenue dans un paquet à part. Par exemple, la documentation du serveur Web Apache est contenue dans le paquet *apache-doc*.

Enfin, lisez les manuels des commandes :

```
% man nom_de_la_commande
```

Note

Souvent, un mini-manuel de la commande est disponible en tapant :

```
% nom_de_la_commande --help
```

36.3 Méthode de résolution des problèmes

Quand vous rencontrez un problème avec un logiciel packagé dans la Debian, les réflexes suivants doivent devenir naturels !

Relire la documentation

Votre premier réflexe doit être de lire - ou plutôt de *relire* - la documentation du logiciel.

Utiliser le Bug Tracking System de Debian

Si votre problème est en fait un bogue du logiciel ou un problème spécifique au paquet Debian, alors il a très probablement déjà été constaté par d'autres utilisateurs de Debian, et il est donc probablement référencé dans le Bug Tracking System (ou BTS) de Debian.

Pour le savoir, allez sur l'interface Web du BTS, accessible à l'adresse bugs.debian.org, et faites une recherche en précisant le nom du paquet et la version de Debian que vous utilisez (*stable*, *testing* ou *unstable*).

Le résultat de votre recherche consistera en une liste de bogues classés par gravité (de *critical* à *wishlist* en passant par *grave*, *serious*, *important*, *normal* et *minor*). Chaque bogue est numéroté et décrit succintement. Si vous cliquez sur un bogue particulier, vous aurez alors une description plus détaillée du problème sous forme d'e-mail (les rapports de bogues et tous les commentaires sont en fait des e-mails), et vous pourrez lire les posts des autres utilisateurs ou développeurs Debian au sujet de ce bogue... la solution à votre problème se trouve peut-être sous vos yeux !

Note

Si vous êtes certain que le problème que vous rencontrez est un bogue, et que ce bogue n'est pas encore référencé dans le BTS, vous pouvez faire un rapport de bogue ! Si vous êtes connecté à Internet, tapez simplement la commande suivante :

```
% reportbug nom_du_paquet
```

et suivez les instructions qui s'affichent dans la console. La lecture de la page Comment signaler un bogue dans Debian vous donnera plus de conseils et de détails sur la procédure à suivre.

Chercher sur le Web

Le Web donne accès à de nombreuses informations. Vous pouvez donc trouver des réponses à vos questions en utilisant un moteur de recherche comme Google ou Exalead, ou un annuaire comme dmoz, alias open directory project.

Ainsi, si vous obtenez un message d'erreur avec un logiciel donné, il suffit généralement de le copier dans un moteur de recherche pour trouver des réponses pertinentes, souvent dans les archives de listes de distribution de projets libres.

Enfin, si vous cherchez de la documentation sur un logiciel particulier, vous pouvez lancer une recherche sur son nom dans dmoz, puis affiner votre recherche en utilisant les catégories. Vous obtiendrez ainsi une liste de sites de qualité sur le sujet de votre choix.

Poster dans les listes de distribution ou les forums

Si, après avoir cherché dans la documentation, dans les FAQ, dans le BTS et dans Google, vous ne trouvez toujours pas de réponse à votre problème, vous pouvez faire appel à l'aide à la communauté Linux.

Trois possibilités s'offrent à vous :

- poster dans les listes de distribution Debian. Il existe des listes sur des sujets techniques particuliers, comme par exemple *debian-laptop* qui traite des aspects techniques spécifiques aux ordinateurs portables, et des listes par langue, comme par exemple *debian-user-french*, où les utilisateurs français de Debian s'entraident. Pour vous abonner aux listes de distribution Debian, rendez-vous à l'adresse www.debian.org/MailingLists/subscribe.
- poster dans les news, par exemple dans le forum *fr.comp.os.linux.configuration*, ou mieux, dans le forum privé de votre école, de votre université ou de votre entreprise consacré à Linux.
- poster dans les listes de distribution des projets OpenSource. Pour connaître leur adresse, rendez-vous sur leur site Web.

Attention

Avant de poster, vérifiez que la réponse à votre question ne se trouve pas dans une FAQ ou dans les archives de la liste ou du forum.

Quatrième partie

Debian GNU/Linux en réseau

Les trois premières parties de cette formation vous ont permis (nous l'espérons !) de savoir refaire avec GNU/Linux ce que vous saviez déjà faire avec Windows.

Dans cette quatrième partie, nous proposons à ceux qui ont un accès permanent à Internet ou qui sont connectés à un réseau local de découvrir une des grandes forces de GNU/Linux : ses capacités de serveur et de client dans un réseau IP.

Note

Dans tous les chapitres qui vont suivre, les logiciels qui établissent une connexion à un serveur peuvent utiliser aussi bien son nom de domaine que son adresse IP. Ainsi, si le serveur *example.org* a pour adresse 192.0.2.166 ou 2001:db8::1:ea:202:a5ff:fecd:13a6 (IPv6 !), ces commandes, que vous découvrirez au Chapitre 40 sont équivalentes :

```
% ssh login@example.org
% ssh login@192.0.2.166
% ssh login@2001:db8:1:ea:202:a5ff:fecd:13a6
```

Chapitre 37

Configuration réseau

Avant de pouvoir utiliser ou fournir des services sur un réseau, un système Debian GNU/Linux doit déjà être connecté à ce réseau. Sur un ordinateur de bureau, cette configuration est souvent automatique, mais il est bon de la maîtriser si l'on souhaite jouer un rôle de serveur.

37.1 Principes

Les réseaux informatiques utilisent un modèle composé de plusieurs couches de protocoles. Nous nous intéressons ici à la troisième couche, dite *couche réseau*, qui utilise le protocole IP (Internet protocol), dans sa version 4 ou 6 : c'est cette couche qui définit la topologie des réseaux, et dont la configuration est par conséquent très importante.

On se connecte à un réseau en utilisant une carte ou une clef réseau. Du point de vue du système d'exploitation, ce périphérique est une *interface réseau*. Sous Linux, ces interfaces sont nommées *eth0, eth1*… pour des interfaces filaires, et *wlan0, wlan1* pour des interfaces sans fil (wifi, wimax…). Il existe également une interface spéciale, nommée *lo* (pour *loopback*) qui désigne toujours votre propre ordinateur.

Note

Dans tout ce chapitre, nous supposerons que vous disposez d'une interface filaire, nommée *eth0*. Nous verrons plus loin comment afficher la liste de vos interfaces.

Éléments de configuration

Une configuration réseau complète, permettant de profiter d'un réseau ou de l'Internet, est constituée des éléments suivants :

une adresse IP cette adresse identifie votre *hôte* sur le réseau où il est connecté ;

un masque de sous-réseau cette donnée indique la partie de votre adresse qui caractérise le réseau local sur lequel votre hôte est connecté, et lui permet de déterminer, pour n'importe quelle adresse IP, si celle-ci fait ou non partie du réseau local ;

une passerelle par défaut c'est l'adresse IP à laquelle il faut transmettre les paquets IP destinés à des hôtes situés hors du réseau local, pour qu'ils soient *routés* vers le réseau local de leur destinataire ;

des serveurs DNS ce sont les adresses de serveurs auxquels votre système ira demander les correspondances entre noms de domaine (*www.debian.org*) et adresses IP (194.109.137.218).

Chaque élément de configuration est nécessaire pour pouvoir utiliser normalement le réseau ou l'Internet :

- sans adresse IP, il est impossible de recevoir les réponses à ses requêtes ;
- sans masque de sous-réseau ou sans passerelle par défaut, il est impossible de communiquer avec les hôtes situés hors du réseau local ;
- sans serveur DNS, on ne peut pas désigner un hôte par son nom de domaine, et il faut donc connaître les adresses IP de tous les serveur que l'on souhaite utiliser

Masque de sous-réseau

Le sous-réseau désigne votre réseau local ou LAN (*local area network*). Il est défini par un *préfixe* d'adresse, par exemple 192.168.0 : toutes les adresses IP qui commencent pas ce préfixe font partie de votre réseau local. Il peut être écrit de deux façon :

par sa longueur en nombre de bits, notée **/longueur** : dans notre exemple, **/24** (chaque chiffre d'une adresse IP fait un octet, soit 8 bits) ;

par un masque semblable à une adresse IP dont tous les bits sont à 1 dans la partie correspondant au préfixe, et à 0 dans la partie restante : dans notre exemple, 255.255.255.0.

Configuration statique

C'est le mode de configuration le plus simple à comprendre : vous devez connaître à l'avance votre configuration complète, pour l'appliquer sur votre système.

Configurer votre connexion consiste alors à affecter à votre carte réseau son adresse IP et son masque de sous-réseau, à ajouter la passerelle par défaut à la table de routage du noyau Linux, et à noter l'adresse des serveurs DNS dans le fichier de configuration du *résolveur DNS*.

Configuration dynamique

Ce mode de configuration, désormais très répandu, est plus adapté aux ordinateurs portables, susceptibles d'être connectés à des réseaux différents, ou aux gens qui ne veulent pas avoir besoin de configurer eux-même leur connexion.

Pour cela, lorsque votre système démarre, ou détecte qu'il vient d'être connecté à un réseau, envoie une demande de paramètres de connexion. Pour un réseau IPv4, cette demande utilise le protocole DHCP (dynamic host configuration protocol) ; pour un réseau IPv6, elle s'effectue dans le cadre d'un processus appelé *découverte de voisinage*, ou par DHCPv6.

Sur un réseau permettant les configurations dynamique, un serveur répond alors en vous proposant une configuration, qui est alors appliquée sur votre système.

37.2 Configuration manuelle

Adresse IP et masque de sous-réseau

La configuration IP proprement dite peut être gérée à l'aide de la commande **ifconfig**. Sans argument, celle-ci affiche la configuration de toutes vos interfaces réseau configurées :

```
% ifconfig
eth0      Link encap:Ethernet  HWaddr 00:1c:23:3f:ff:bb
          inet adr:192.168.0.105  Bcast:192.168.0.255  Masque ↩
              :255.255.255.0
          adr inet6: fe80::21c:23ff:fe3f:ffbb/64 Scope:Lien
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:749880 errors:0 dropped:0 overruns:0  ↩
              frame:0
          TX packets:393902 errors:0 dropped:0 overruns:0  ↩
              carrier:0
          collisions:0 lg file transmission:1000
          RX bytes:910931782 (868.7 MiB)  TX bytes:32422248  ↩
              (30.9 MiB)
          Interruption:17

lo        Link encap:Boucle locale
          inet adr:127.0.0.1  Masque:255.0.0.0
          adr inet6: ::1/128 Scope:Hôte
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:71 errors:0 dropped:0 overruns:0 frame:0
          TX packets:71 errors:0 dropped:0 overruns:0 carrier ↩
              :0
          collisions:0 lg file transmission:0
          RX bytes:10096 (9.8 KiB)  TX bytes:10096 (9.8 KiB)
```

Note

Cette commande peut également afficher les interfaces non configurées, avec l'argument **-a**, pour *all*. Cela peut servir à déterminer le nom d'une interface non encore configurée.

Pour appliquer une configuration à une interface, on utilise la même commande, avec des arguments correspondants à cette configuration. Par exemple, pour utiliser l'adresse IPv4 192.168.0.42 et le masque de sous-réseau 255.255.255.0 sur l'interface *eth0*, en tant que root, tapez :

```
# ifconfig eth0 192.168.0.42
```

Pour ajouter l'adresse IPv6 2001:db8::6726 avec un sous-réseau de longueur 32, tapez :

```
# ifconfig eth0 add 2001:db8::6726/32
```

Note

Pour supprimer une configuration IPv4, utilisez la commande **ifconfig eth0 down**. Pour supprimer une configuration IPv6, utilisez la commande **ifconfig eth0 del <adresse>**.

Passerelle par défaut

La passerelle par défaut permet de définir l'hôte de votre réseau local vers lequel envoyer tous les paquets destinés à des hôtes situés hors du réseau local, ce qui définit une *route par défaut*. Cette route doit être ajoutée à la table de routage du noyau Linux, avec la commande **route**. Par exemple, si votre passerelle par défaut a pour adresse IPv4 192.168.0.1, et peut être jointe par votre interface *eth0*, ajoutez ainsi la route par défaut :

```
# route add default gw 192.168.0.1 dev eth0
```

Si vous êtes sur un réseau IPv6 qui possède une passerelle 2001:db8::1, utilisez l'option **--inet6**, qui permet de manipuler la table de routage IPv6 plutôt que la table IPv4 :

```
# route --inet6 add default gw 2001:db8::1 dev eth0
```

Pour afficher la table de routage conmplète, utilisez la commande **route** sans arguments. Pour supprimer une route, utilisez la commande **route del**, suivie des mêmes arguments :

```
% route --inet6
Table de routage IPv6 du noyau
Destination                    Next Hop                    ↩
    Flag Met Ref Use If
2001::/32                      ::                         U  ↩
       256 0     1 eth0
fe80::/64                      ::                         U  ↩
       256 0     0 eth0
::/0                           2001:db8::1                UG  ↩
      1   0      0 eth0
ff00::/8                       ::                         U  ↩
       256 0     0 eth0
# route --inet6 del default gw 2001:db8::1 dev eth0
```

Note

La configuration IP et la table de routage peuvent être manipulées de façon plus avancée par une commande unique, **ip**, disponible dans le paquet *iproute*, dont vous pouvez consulter le site web.

Serveurs DNS

La configuration IP et la route par défaut concernaient le noyau Linux. Les serveurs DNS concernent le *résolveur*, une fonction logicielle qui permet de traduire les noms de domaines en adresses IP. Ce résolveur utilise pour cela les serveurs dont les adresses

sont notées dans le fichier `/etc/resolv.conf`. Si le serveur DNS de votre fournisseur d'accès, a pour adresse IPv4 192.0.2.71, éditez ce fichier pour qu'il contienne :

```
nameserver 192.0.2.71
```

Configuration dynamique

Dans un réseau IPv4, vous pouvez demander une configuration par DHCP en utilisant le programme **dhclient** :

```
# dhclient eth0
```

Si un serveur DHCP est disponible sur votre réseau et répond à votre requête, **dhclient** vous rend la main en affichant des informations sur la configuration obtenue, appelée *bail DHCP*. Dans le cas contraire, il continue d'envoyer des requêtes DHCP indéfiniment, jusqu'à ce que vous décidiez de l'interrompre en appuyant sur Ctrl-c. Vous devrez alors procéder à une configuration manuelle...

37.3 Configuration permanente

Évidemment, il serait hors de question de devoir saisir à la main les commandes permettant de configurer votre connexion à chaque fois que vous démarrez votre système. Debian permet donc d'enregistrer votre configuration dans un fichier, `/etc/network/interfaces` :

```
auto lo eth1
allow-hotplug eth0

iface lo inet loopback

iface eth0 inet dhcp

iface eth1 inet static
    address 192.168.0.42
    netmask 255.255.255.0
    gateway 192.195.0.1
    dns-nameservers 192.0.2.71

iface eth1 inet6 static
    address 2001:db8::6726
    netmask 32
    gateway 2001:db8::1
    dns-nameservers 2001:db8::12
```

Interfaces à configurer automatiquement

Les instructions **auto**, qui peuvent se trouver plusieurs fois et n'importe où dans le fichier, précisent quelles interfaces doivent être configurées au démarrage du système d'exploitation. On les place en général juste avant la définition de chaque interface.

Les instructions **allow-hotplug** ont le même effet, à ceci près que l'interface ne sera configurée au démarrage *que si* elle est connectée à in réseau.

Dans mon exemple, les interfaces *lo* et *eth1* seront donc configurées au démarrage, qu'elles soient branchées ou non. L'interface *eth0*, quant à elle, sera configurées au démarrage si elle est branchée.

Configuration d'une interface

La configuration de chaque interface est décrite dans une section **iface <nom_de_l'interface>**. Celle-ci est immédiatement suivie du protocole utilisé : **inet** pour une configuration IPv4 et **inet6** pour une configuration IPv6. Enfin, toujours sur la même ligne, on trouve le type de configuration à utiliser : **static**, ou **dhcp** pour une configuration dynamique par DHCP.

Note

Pour une configuration IPv6 dynamique, il est inutile de préciser une configuration dans le fichier `interfaces` : le protocole IPv6 a été conçu de telle façon que le processus découverte de voisinage est entièrement pris en charge par le noyau.

Les interfaces configurées par DHCP n'ont besoin d'aucune autre paramètre. En revanche, pour les interfaces dont la configuration est définie de façon statique, il faut préciser cette configuration :

address l'adresse IP attribuée à l'interface ;

netmask pour une configuration IPv4, le masque de sous-réseau ; pour une configuration IPv6, la longueur du préfixe correspondant au sous-réseau ;

gateway l'adresse IP de la passerelle ;

dns-nameservers les adresses IP de vos serveurs DNS.

Note

La directive **dns-nameservers** ne peut être utilisée qu'après avoir installé le paquet *resolvconf*, qui n'est *pas* préinstallé avec Debian.

Note

Pour plus de détails sur ce fichier de configuration, vous pouvez consulter son manuel : **man interfaces**.

ASTUCE

Il est possible d'attribuer plusieurs configurations IP à une même adresse. La façon la plus simple pour cela est de définir des interfaces virtuelles. À partir de l'interface *eth0*, vous pouvez définir les interfaces *eth0:0, eth0:1*..., qui seront alors utilisable comme des interfaces réelles.

Appliquer la configurations d’une interface

Les interfaces listée comme **auto** ou **allow-hotplug** sont activées selon leur configuration au démarrage de Debian. Si vous venez de rédiger la configurations d’une interface, ou si vous voulez l’activer ou la désactiver à la main, utilisez les commandes **ifup** (pour l’activer) et **ifdown** (pour la désactiver) :

```
# ifup eth0
```

Chapitre 38

Pare-feu et partage de connexion Internet

Important
Cette partie requiert des connaissances de base en réseau. Vous pouvez consulter une formation VIA à ce sujet.

Le partage de connexion Internet se fait sous Linux grâce aux fonctions de filtrage du noyau... d'où le regroupement des explications sur le filtrage et sur le partage de connexion Internet dans ce chapitre !

38.1 Le partage de connexion Internet

L'idée est d'ajouter à votre ordinateur sous Linux la fonction de *routeur NAT*, qui va vous permettre de partager votre connexion avec d'autres machines (Figure 38.1 et Figure 38.2). Votre ordinateur sous Linux aura deux interfaces réseau :

Important
Si vous êtes connecté derrière une *box fournie par votre FAI, celle-ci joue déjà ce rôle de routeur NAT. Plutôt que de configurer votre ordinateur, vous devez alors utiliser l'interface d'administration de votre modem routeur pour y définir la configuration réseau souhaitée.

– une interface connectée à Internet via une connexion modem par exemple : cette interface aura une adresse IP publique (interface *eth0* sur les schémas) ;
– une interface connectée à votre réseau local doté d'un adressage privé : cette interface servira de passerelle pour les ordinateurs du réseau local (interface *eth1* sur les schémas).

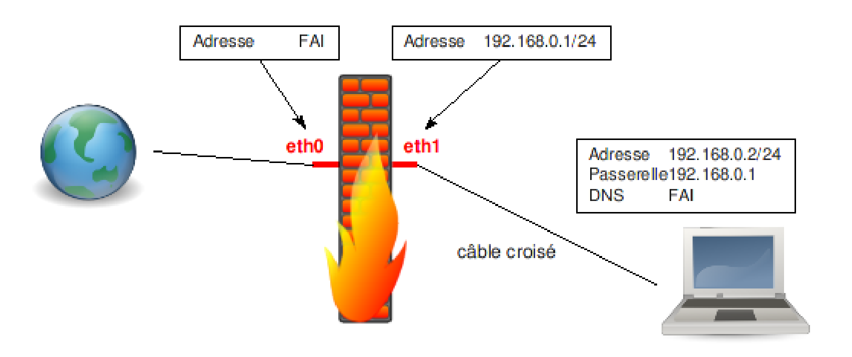


FIGURE 38.1 – Schéma d'un NAT avec 2 machines

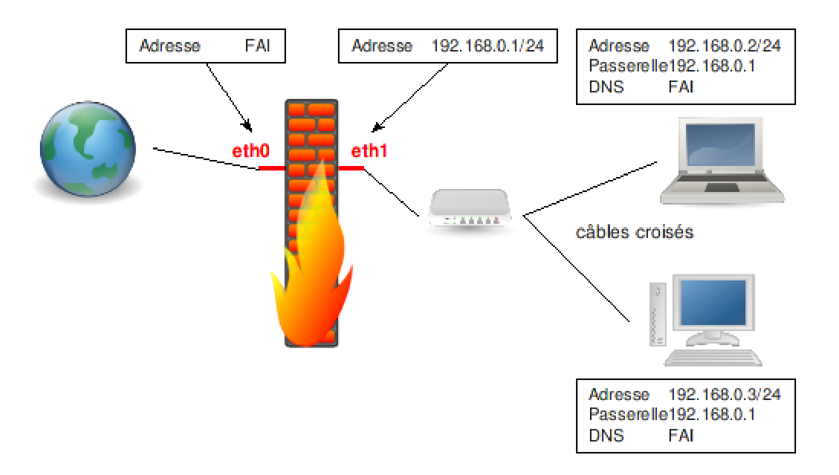


FIGURE 38.2 – Schéma d'un NAT avec plusieurs machines

Avec cette configuration, seul le routeur NAT est directement joignable depuis Internet ; les ordinateurs du réseau local ne sont a priori pas joignables directement depuis Internet... sauf si on fait du *transfert de port*. Par exemple, pour que le serveur Web qui tourne sur le port 80 de la machine dont l'adresse IP est 192.168.0.3 soit joignable de l'extérieur, il faut transférer les requêtes TCP arrivant sur le port 80 du routeur NAT vers la machine dont l'adresse est 192.168.0.3, et la réponse à la requête sera alors correctement transférée vers Internet au client qui a initié la requête.

Note

Pour améliorer les temps de réponses des requêtes DNS, il pourra être intéressant de mettre en place un serveur DNS de type *indépendant* sur le routeur NAT, comme expliqué dans le wiki, et d'indiquer dans la configuration réseau des ordinateurs du réseau local l'adresse 192.168.0.1 comme premier serveur DNS.

38.2 Etablir des règles de filtrage et de partage de connexion

Apprendre la syntaxe iptables

La syntaxe d'*iptables* est très complète... et je n'ai malheureusement pas l'intention de l'expliquer dans cette annexe. Je l'ai personnellement apprise dans le Hors Série Linux Magazine n°12 dédié aux pare-feux. Si vous maîtrisez l'anglais, vous pouvez vous attaquer au tutoriel iptables.

Configurer iptables

Le paquet *iptables*, qui contient le programme du même nom qui permet de configurer les fonctions de filtrage IP des noyaux Linux 2.4 et 2.6, est normalement installé par défaut.

Nous allons mettre en place deux scripts à l'aide de mes fichiers d'exemple :

– le script `/etc/network/iptables.up.rules` qui définit les règles de filtrage IPv4 :

```
# cp ~/fichiers-config/iptables.up.rules /etc/
```

ou :

```
% wget http://formation-debian.via.ecp.fr/fichiers- ↩
    config/iptables.up.rules
# cp iptables.up.rules /etc/
```

– le script `/etc/network/ip6tables.up.rules` qui définit les règles de filtrage IPv6 :

```
# cp ~/fichiers-config/ip6tables.up.rules /etc/
```

ou :

```
% wget http://formation-debian.via.ecp.fr/fichiers- ↩
    config/ip6tables.up.rules
# cp ip6tables.up.rules /etc/
```

Il faut maintenant demander de charger ces définitions à l'initialisation du réseau de votre système. Éditez pour cela le fichier `/etc/network/interfaces`, et ajoutez ces deux lignes à la définition de l'interface connectée à Internet :

```
iface eth0 ...
    ...
    pre-up iptables-restore < /etc/iptables.up.rules
    pre-up ip6tables-restore < /etc/ip6tables.up.rules
```

Vous pouvez également démarrer le filtrage en chargeant ces définitions à la main, en tant que root, avec la commande **ip[6]tables-restore < /etc/ip[6]tables.up.rules**.

Personnaliser les règles de filtrage

Personnalisez mes définitions d'exemple `/etc/ip[6]tables.up.rules` qui contiennent les règles de filtrage :

```
# /etc/network/iptables.up.rules
# Script qui démarre les règles de filtrage IPv4
# Formation Debian GNU/Linux par Alexis de Lattre
# http://formation-debian.via.ecp.fr/

# iptables-restore(8) remet implicitement à zéro toutes les ↩
    règles

# Les instructions qui suivent concernent la table «~filter~» ↩
    ,
# c'est à dire... le filtrage.
*filter

#########################
# Politiques par défaut #
#########################
# Les politiques par défaut déterminent le devenir d'un ↩
    paquet auquel
# aucune règle spécifique ne s'applique.

# Les connexions entrantes sont bloquées par défaut
-P INPUT DROP
# Les connexions destinées à être routées sont acceptées par ↩
    défaut
-P FORWARD ACCEPT
# Les connexions sortantes sont acceptées par défaut
-P OUTPUT ACCEPT

######################
# Règles de filtrage #
######################
# Nous précisons ici des règles spécifiques pour les paquets ↩
    vérifiant
# certaines conditions.

# Pas de filtrage sur l'interface de "loopback"
-A INPUT -i lo -j ACCEPT

# Accepter le protocole ICMP (notamment le ping)
-A INPUT -p icmp -j ACCEPT

# Accepter le protocole IGMP (pour le multicast)
-A INPUT -p igmp -j ACCEPT

# Accepter les packets entrants relatifs à des connexions ↩
    déjà
# établies~: cela va plus vite que de devoir réexaminer ↩
    toutes
# les règles pour chaque paquet.
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT

# Décommentez la ligne suivante pour que le serveur SSH ↩
    éventuel
```

```
# soit joignable de l'extérieur
#-A INPUT -p tcp --dport ssh -j ACCEPT

# Décommentez les lignes suivantes pour que le serveur de ↩
    courrier
# éventuel soit joignable de l'extérieur.
#-A INPUT -p tcp --dports smtp -j ACCEPT
# Si vous avez activé les services SMTPS et soumission de ↩
    messages...
#-A INPUT -p tcp --dports smtps -j ACCEPT
#-A INPUT -p tcp --dports submission -j ACCEPT

# Décommentez les deux lignes suivantes pour que le serveur ↩
    de noms
# éventuel soit joignable de l'extérieur.
#-A INPUT -p tcp --dport domain -j ACCEPT
#-A INPUT -p udp --dport domain -j ACCEPT

# Décommentez la ligne suivante pour que le serveur Web ↩
    éventuel
# soit joignable de l'extérieur.
#-A INPUT -p tcp --dport http -j ACCEPT
# Si vous avez activé le HTTPS...
#-A INPUT -p tcp --dport https -j ACCEPT

# Décommentez les deux lignes suivantes pour que le serveur d ↩
    'impression
# éventuel soit joignable de l'extérieur.
#-A INPUT -p tcp --dport ipp -j ACCEPT
#-A INPUT -p udp --dport ipp -j ACCEPT

# Décommentez les deux lignes suivantes pour que le serveur ↩
    Samba
# éventuel soit joignable de l'extérieur.
#-A INPUT -p tcp --dport netbios-ssn -j ACCEPT
#-A INPUT -p udp --dport netbios-ssn -j ACCEPT

# Décommentez la ligne suivante pour que des clients puissent ↩
     se connecter
# à l'ordinateur par XDMCP.
#-A INPUT -p udp --dport xdmcp -j ACCEPT

# Décommentez la ligne suivante pour que l'ordinateur puisse ↩
    se connecter
# par XDMCP à une machine distante).
#-A INPUT -p tcp --dport x11-1 -j ACCEPT

# Décommentez la ligne suivante pour pouvoir recevoir des ↩
    flux VideoLAN.
#-A INPUT -p udp --dport 1234 -j ACCEPT

# Décommentez la ligne suivante pour pouvoir recevoir des ↩
    annonces SAP
# (ce sont des annonces de session multicast).
```

```
#-A INPUT -p udp -d 224.2.127.254 --dport 9875 -j ACCEPT

# Décommentez les 3 lignes suivantes pour pouvoir utiliser ↩
    GnomeMeeting
#-A INPUT -p tcp --dport 30000:33000 -j ACCEPT
#-A INPUT -p tcp --dport 1720 -j ACCEPT
#-A INPUT -p udp --dport 5000:5006 -j ACCEPT

# Décommentez la ligne suivante pour pouvoir partager de la ↩
    musique par
# DAAP.
#-A INPUT -p tcp --dport daap -j ACCEPT

# Décommentez la ligne suivante pour que votre ordinateur
# annonce son nom et ses services par mDNS sur le réseau ↩
    local (cela
# permet de le contacter sous «~son nom d'hôte~».local).
-A INPUT -p udp -d 224.0.0.251 --dport mdns -j ACCEPT

# La règle par défaut pour la chaine INPUT devient REJECT ( ↩
    contrairement
# à DROP qui ignore les paquets, avec REJECT, l'expéditeur ↩
    est averti
# du refus). Il n'est pas possible de mettre REJECT comme ↩
    politique par
# défaut. Au passage, on note les paquets qui vont être jetés ↩
    , ça peut
# toujours servir.
-A INPUT -j LOG --log-prefix "paquet IPv4 inattendu "
-A INPUT -j REJECT

COMMIT

# Les instructions qui suivent concernent la table «~nat~».
*nat

########################
# Partage de connexion #
########################

# Décommentez la ligne suivante pour que le système fasse ↩
    office de
# routeur NAT et remplacez «~eth0~» par le nom de l'interface
# connectée à Internet.
#-A POSTROUTING -o eth0 -j MASQUERADE


########################
# Redirections de port #
########################

# Décommentez la ligne suivante pour que les requêtes TCP ↩
    reçues sur
# le port 80 de l'interface eth0 soient redirigées à la ↩
```

```
    machine dont
# l'adresse IPv4 est 192.168.0.3 sur son port 80 (la réponse  ↩
    à la
# requête sera transférée au client).
#-A PREROUTING -i eth0 -p tcp --dport 80 -j DNAT --to- ↩
    destination 192.168.0.3:80

COMMIT

####################
# Problème de MTU... #
####################

# Les instructions qui suivent concernent la table «~mangle~» ↩
    , c'est
# à dire l'altération des paquets
*mangle

# Si la connexion que vous partagez est une connexion ADSL  ↩
    directement gérée
# par votre ordinateur, vous serez probablement confronté au  ↩
    fameux problème du
# MTU. En résumé, le problème vient du fait que le MTU de la  ↩
    liaison entre
# votre fournisseur d'accès et le serveur NAT est un petit  ↩
    peu inférieur au MTU
# de la liaison Ethernet qui relie le serveur NAT aux  ↩
    machines qui sont
# derrière le NAT. Pour résoudre ce problème, décommentez la  ↩
    ligne suivante et
# remplacez «~eth0~» par le nom de l'interface connectée à  ↩
    Internet.
#-A FORWARD -p tcp --tcp-flags SYN,RST SYN -j TCPMSS -o eth0  ↩
    --clamp-mss-to-pmtu

COMMIT
```

Configurer le réseau pour le partage de connexion

Si vous désirez mettre en place un partage de connexion Internet, il faut commencer par bien configurer les interfaces réseau du routeur NAT. Par exemple, pour un routeur NAT dont l'interface réseau connectée au réseau extérieur est *eth0* avec configuration par DHCP et dont l'interface connectée au réseau local est *eth1*, le fichier `/etc/network/interfaces` doit ressembler à l'exemple suivant :

```
# /etc/network/interfaces
# Fichier de configuration d'exemple des interfaces réseau
# pour faire un serveur NAT
# Formation Debian GNU/Linux par Alexis de Lattre
# http://formation-debian.via.ecp.fr/

# Plus d'informations dans "man interfaces"
```

```
# L'interface "loopback"
auto lo
iface lo inet loopback

# L'interface "eth0" connectée à Internet (configuration par ↩
    DHCP)
auto eth0
iface eth0 inet dhcp

# L'interface "eth1" connectée au réseau local (IP privée ↩
    fixe)
auto eth1
iface eth1 inet static
    address 192.168.0.1
    netmask 255.255.255.0
    broadcast 192.168.0.255
```

Enfin, il faut activer la fonction de transfert IP du noyau, en modifiant ainsi le fichier `/etc/sysctl.conf` :

```
# Uncomment the next line to enable packet forwarding for ↩
    IPv4
net.ipv4.ip_forward=1
# Uncomment the next line to enable packet forwarding for ↩
    IPv6
net.ipv6.conf.all.forwarding=1
```

Ces paramètres seront appliqués au prochain démarrage. Pour les appliquer dès maintenant :

```
# sysctl -p
```

Démarrer le filtrage ou le partage de connexion

Une fois que vous avez bien configuré le fichier `/etc/network/interfaces` et personnalisé le script `/etc/iptables.up.rules` selon vos besoins, demandez au système de recharger ces règles :

```
    # iptables-restore < /etc/iptables.up.rules
```

Afficher la configuration iptables

Pour afficher la configuration iptables actuelle, tapez :

– pour la table *filter* :

```
# iptables -v -L
```

– pour la table *nat* :

```
# iptables -v -L -t nat
```

Chapitre 39

Monter un pont réseau

Important

Cette partie requiert des connaissances de base en réseau Ethernet. Vous pouvez consulter une formation VIA à ce sujet.

39.1 Principe

L'idée est d'ajouter à votre ordinateur sous Linux la fonction de *commutateur Ethernet* aussi appelée *pont* ou *bridge*.

Pour ajouter à votre ordinateur sous Linux cette fonction de *pont*, il lui faut plusieurs cartes réseau, qui seront utilisées comme ports du commutateur, comme sur la Figure 39.1. Ce *pont* fonctionnera comme un commutateur Ethernet classique, en apprenant les adresses MAC qui sont derrière ses interfaces réseau pour aiguiller les trames Ethernet. En revanche, contrairement à un commutateur classique, il ne croisera pas la connexion réseau : il faudra donc relier le *bridge* aux autres ordinateurs par des câbles *croisés*, et aux autres commutateurs par des câbles *droits* (les câbles normaux sont des câbles droits).

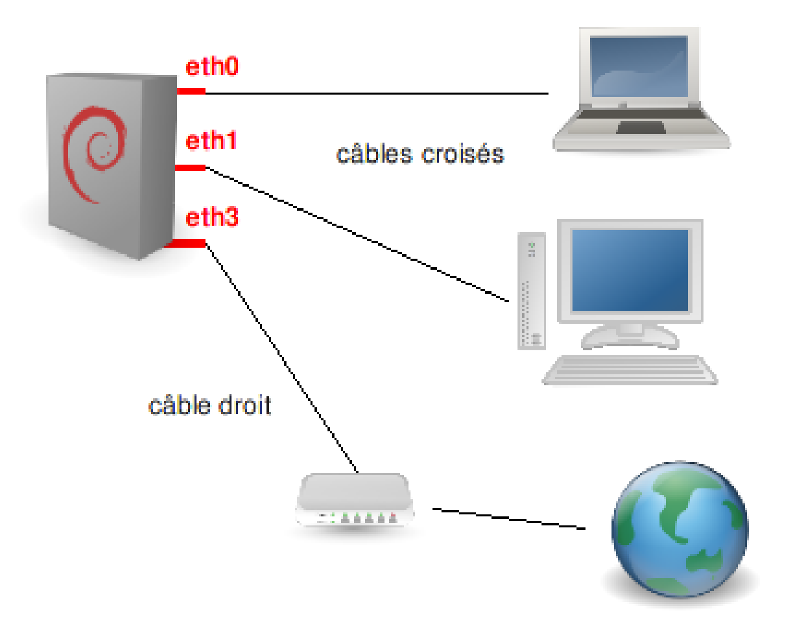


FIGURE 39.1 – Schéma d'un pont

39.2 Configuration

Installer le programme de gestion des ponts réseau :

```
# aptitude install bridge-utils
```

Lancer le bridge au démarrage

La définition d'un pont se fait normalement en utilisant la commande **brctl**. Pour que la configuration soit appliquée à chaque démarrage, nous allons en faitmodifier le fichier de configuration des interfaces réseau `/etc/network/interfaces` en utilisant mon fichier de configuration d'exemple et en le personnalisant :

```
# mv /etc/network/interfaces /etc/network/interfaces.old
# cp ~/fichiers-config/interfaces-bridge /etc/network/ ↩
    interfaces
```

ou :

```
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
    interfaces-bridge
# mv /etc/network/interfaces /etc/network/interfaces.old
# mv interfaces-bridge /etc/network/interfaces
```

Note

Lorsque vous définissez un pont, les interfaces utilisées comme ports deviennent inutilisables au niveau réseau. En revanche, Linux définit une interface virtuelle (ici, *br0*), qui peut être configurée et utilisée sur l'ordinateur comme si elle était connectée au pont réseau.

Cette interface reçoit l'adresse MAC la plus faible parmi les interfaces utilisées comme ports, qui sont lues comme des nombres hexadécimaux. Ainsi, si votre pont relie les interfaces *eth0* 00:0c:6e:b1:c7:95, *eth1* 00:08:54:3d:cf:bb et *eth3* 00:05:5d:a1:e1:ad, l'interface virtuelle *br0* prendra l'adresse MAC 00:05:5d:a1:e1:ad (son troisième chiffre, 5 étant inférieur à celui des deux autres interfaces, c et 8).

Personnalisez le nouveau fichier `/etc/network/interfaces` ; les lignes de commentaire doivent vous permettre de comprendre chaque paramètre :

```
# /etc/network/interfaces
# Fichier de configuration d'exemple des interfaces réseau
# pour faire un bridge
# Formation Debian GNU/Linux par Alexis de Lattre
# http://formation-debian.via.ecp.fr/

# Plus d'informations dans "man interfaces", "man bridge- ↩
   utils-interfaces" et
# dans "/usr/share/doc/bridge-utils/README.Debian.gz"

# D'abord l'interface "loopback"
auto lo
iface lo inet loopback

# Ensuite l'interface bridge "br0" qui se configure par DHCP
auto br0
iface br0 inet dhcp
    # Liste des interfaces qui participent au bridge
    # ATTENTION :
    # Il faut mettre l'interface dont la MAC est la plus  ↩
       petite d'abord !
    # Sinon, cela peut perturber les outils de surveillance  ↩
       du réseau.
    bridge_ports eth0 eth1 eth2
    # Je désactive le Spanning tree
    bridge_stp off
    # Temps en secondes entre "learning state" et "forwarding ↩
        state"
    bridge_fd 2
    # Temps maximum en secondes où le script de lancement du  ↩
       bridge
    # attendra lors du démarrage que le bridge passe en mode  ↩
       "forwarding
    # state" pour passer la main et laisser les autres  ↩
       services démarrer.
    bridge_maxwait 0
```

Relancez la configuration des interfaces réseau :

```
# /etc/init.d/networking restart
Reconfiguring network interfaces: done.
```

Pendant ce redémarrage, regardez votre console de log (si vous avez suivi la formation depuis le début, c'est la huitième console) ou faites apparaître la fin du *syslog* en direct avec la commande :

```
% tail -f /var/log/syslog
```

Vous verrez alors 3 étapes dans la lancement du bridge :

1. *listening* : il regarde sur les interfaces réseau physiques les packets qui arrivent,
2. *learning* : il en déduit la configuration du réseau tout seul, notamment quelles adresses MAC sont derrière quels ports,
3. *forwarding* : le *bridge* se met à fonctionner, le réseau marche enfin !

Chapitre 40

L'accès à distance par SSH

40.1 Introduction et mise-en-garde

Qu'est-ce que SSH ?

SSH signifie *Secure SHell*. C'est un protocole qui permet de faire des connexions sécurisées (i.e. chiffrées) entre un serveur et un client SSH. Nous allons utiliser le programme OpenSSH, qui est la version libre du client et du serveur SSH.

Mise en garde sur la sécurité

Nature du problème

Installer un serveur SSH permet aux utilisateurs d'accéder au système à distance, en rentrant leur login et leur mot de passe (ou avec un mécanisme de clefs). Cela signifie aussi qu'un pirate peut essayer d'avoir un compte sur le système (pour accéder à des fichiers sur le système ou pour utiliser le système comme une passerelle pour attaquer d'autres systèmes) en essayant plein de mots de passes différents pour un même login (il peut le faire de manière automatique en s'aidant d'un dictionnaire électronique). On appelle ça une attaque *en force brute*.

Il y a donc trois contraintes majeures pour garder un système sécurisé après avoir installé un serveur SSH :

- avoir un serveur SSH à jour au niveau de la sécurité, ce qui doit être le cas si vous faites consciencieusement les mises à jour de sécurité en suivant la procédure Debian, comme expliqué au Chapitre 21 ;
- que les mots de passes de *TOUS* les utilisateurs soient suffisamment complexes pour résister à une attaque en force brute ;
- surveiller les connexions en lisant régulièrement le fichier de log `/var/log/auth.log`.

Choisir des mots de passe complexes

Un mot de passe complexe est un mot de passe qui ne veut rien dire, qui n'est pas dans le dictionnaire et qui comporte au moins 8 caractères, de préférence avec un mélange de lettres minuscules, de lettres majuscules, de chiffres et de caractères de ponctuation.

Une bonne méthode pour obtenir un mot de passe complexe et facile à retenir consiste à choisir une phrase et à prendre la première lettre de chaque mot, avec quelques complications en plus.

Par exemple, la phrase « *Linux, moi j'y comprends rien de rien !* » donne le mot de passe **L,mj'ycr2r!**

Tester la complexité des mots de passe

Pour vérifier que les mots de passe des utilisateurs du système sont vraiment complexes, le root peut les soumettre à un cracker de mots de passe... et voir combien de temps ils résistent !

Les mots de passes des utilisateurs sont stockés dans le fichier `/etc/shadow`. Seul l'utilisateur root peut lire ce fichier. Pour tester la complexité des mots de passes, root peut donc installer le programme john et le lancer sur le fichier `/etc/shadow` :

```
# aptitude install john
# john /etc/shadow
```

Quand *john* a trouvé un mot de passe, il l'affiche avec le login associée.

Attention, *john* utilisera le processeur à 100 % ! Il est donc conseillé de lui donner un priorité faible (commande **nice** ou **renice**) si la machine doit être utilisée pendant ce temps. Plus le nombre d'utilisateurs est grand, plus il faudra laisser tourner *john* longtemps pour que le test soit significatif.

40.2 Le système de clefs de SSH

La théorie de la cryptographie asymétrique

SSH utilise la cryptographie asymétrique RSA ou DSA. En cryptographie asymétrique, chaque personne dispose d'un couple de clef : une clé publique et une clef privée. La clé publique peut être librement publiée tandis que la clef privée doit rester secrète. La connaissance de la clef publique ne permet pas d'en déduire la clé privée.

Si Alice veut envoyer un message confidentiel à Bob, elle doit le chiffrer avec la clef publique de Bob et lui envoier sur un canal qui n'est pas forcément sécurisé. Seul Bob pourra déchiffrer ce message en utilisant sa clef privée.

La théorie de la cryptographie symétrique

SSH utilise également la cryptographie symétrique. Son principe est simple : si Alice veut envoyer un message confidentiel à Bob, Alice et Bob doivent d'abord posséder une même clef secrète. Alice chiffre le message avec la clé secrète puis l'envoie à Bob sur un canal qui n'est pas forcément sécurisé. Bob déchiffre alors le message grâce à la clef secrète. Toute autre personne en possession de la clef secrète peut également déchiffrer le message.

Les algorithmes de chiffrement symétrique sont beaucoup moins gourmands en ressources processeur que ceux de chiffrement asymétrique... mais le gros problème est l'échange de la clef secrète entre Alice et Bob. Dans le protocole SSL, qui est utilisé par SSH et par les navigateurs Web, la cryptographie asymétrique est utilisée au début de la communication pour que Alice et Bob puissent s'échanger une clef secrète de manière sécurisée, puis la suite la communication est sécurisée grâce à la cryptographie symétrique en utilisant la clef secrète ainsi échangée.

L'établissement d'une connexion SSH

Un serveur SSH dispose d'un couple de clefs RSA stocké dans le répertoire `/etc/ssh/` et généré lors de l'installation du serveur. Le fichier `ssh_host_rsa_key` contient la clef privée et a les permissions 600. Le fichier `ssh_host_rsa_key.pub` contient la clef publique et a les permissions 644.

Nous allons suivre par étapes l'établissement d'une connexion SSH :

1. Le serveur envoie sa clef publique au client. Celui-ci vérifie qu'il s'agit bien de la clef du serveur, s'il la déjà reçue lors d'une connexion précédente.
2. Le client génère une clef secrète et l'envoie au serveur, en chiffrant l'échange avec la clef publique du serveur (chiffrement asymétrique). Le serveur déchiffre cette clef secrète en utilisant sa clé privée, ce qui prouve qu'il est bien le vrai serveur.
3. Pour le prouver au client, il chiffre un message standard avec la clef secrète et l'envoie au client. Si le client retrouve le message standard en utilisant la clef secrète, il a la preuve que le serveur est bien le vrai serveur.
4. Une fois la clef secrète échangée, le client et le serveur peuvent alors établir un canal sécurisé grâce à la clef secrète commune (chiffrement symétrique).
5. Une fois que le canal sécurisé est en place, le client va pouvoir envoyer au serveur le login et le mot de passe de l'utilisateur pour vérification. La canal sécurisé reste en place jusqu'à ce que l'utilisateur se déconnecte.

La seule contrainte est de s'assurer que la clef publique présentée par le serveur est bien sa clef publique... sinon le client risque de se connecter à un faux serveur qui aurait pris l'adresse IP du vrai serveur (ou toute autre magouille). Une bonne méthode est par exemple de demander à l'administrateur du serveur quelle est le *fingerprint* de la clef publique du serveur avant de s'y connecter pour la première fois. Le *fingerprint* d'une clef publique est une chaîne de 32 caractères hexadécimaux à peu près unique pour chaque clef (un hachage) ; il s'obtient grâce à la commande **ssh-keygen -lf /etc/ssh/ssh_host_rsa_key.pub**.

40.3 Installation et configuration de SSH

Installation du serveur SSH

Le client SSH est disponible dans le paquet *openssh-client*, qui est préinstallé.

Pour pouvoir vous connecter à distance, vous pouvez maintenant installer le serveur SSH :

```
# aptitude install openssh-server
```

L'installation comporte une étape de génération des clefs de chiffrement. Finalement, le serveur SSH se lance.

Configuration du serveur SSH

Le fichier de configuration du serveur SSH est `/etc/ssh/sshd_config`. À ne pas confondre avec le fichier `/etc/ssh/ssh_config`, qui est le fichier de configuration du client SSH.

Nous allons vous commenter les lignes les plus importantes de ce fichier de configuration :

- ```
  Port 22
  ```

  Signifie que le serveur SSH écoute sur le port 22, qui est le port par défaut de SSH. Vous pouvez le faire écouter sur un autre port en changeant cette ligne. Vous pouvez aussi le faire écouter sur plusieurs ports à la fois en rajoutant des lignes similaires.
- ```
  PermitRootLogin yes
  ```

  Signifie que vous pouvez vous connecter en root par SSH. Vous pouvez changer et mettre *no*, ce qui signifie que pour vous connecter en root à distance, vous devrez d'abord vous connecter par SSH en tant que simple utilisateur, puis utiliser la commande **su** pour devenir root. Sans cela, un pirate n'aurait qu'à trouver le mot de passe du compte root, alors que là, il doit trouver votre login *et* votre mot de passe.
- ```
  X11Forwarding yes
  ```

  Signifie que vous allez pouvoir travailler en déport d'affichage par SSH. Ce sera expliqué plus tard, dans le Chapitre 42.

Si vous avez modifié le fichier de configuration du serveur, il faut lui dire de relire son fichier de configuration :

```
# /etc/init.d/ssh reload
Reloading OpenBSD Secure Shell server's configuration.
```

40.4 Se connecter par SSH

Authentification par mot de passe

C'est la méthode la plus simple. Depuis la machine cliente, tapez :

```
% ssh login@nom_DNS_du_serveur_SSH
```

ASTUCE

Si vous utilisez le même identifiant sur le client et sur le serveur, vous pouvez vous contenter de taper :

```
% ssh nom_DNS_du_serveur_SSH
```

- Si c'est la première connexion SSH depuis ce client vers ce serveur, il vous demande si le fingerprint de la clef publique présentée par le serveur est bien le bon. Pour être sûr que vous vous connectez au bon serveur, vous devez connaître de façon certaine le fingerprint de sa clef publique et la comparer à celle qu'il vous affiche. Si les deux fingerprints sont identiques, répondez *yes*, et la clef publique du serveur est alors rajoutée au fichier `~/.ssh/known_hosts`.
- Si vous vous êtes déjà connecté depuis ce client vers le serveur, sa clef publique est déjà dans le fichier `~/.ssh/known_hosts` et il ne vous demande donc rien.

Ensuite, entrez votre mot de passe... et vous verrez apparaître le prompt, comme si vous vous êtiez connecté en local sur la machine.
```

Authentification par clef

Au lieu de s'authentifier par mot de passe, les utilisateurs peuvent s'authentifier grâce à la cryptographie asymétrique et son couple de clefs privée/publique, comme le fait le serveur SSH auprès du client SSH.

Générer ses clefs

Pour générer un couple de clefs DSA, tapez :

```
% ssh-keygen -t dsa
```

Les clefs générées ont par défaut une longueur de 1024 bits, ce qui est aujourd'hui considéré comme suffisant pour une bonne protection.

La clef privée est stockée dans le fichier `~/.ssh/id_dsa` avec les permissions 600 et la clef publique est stockée dans le fichier `~/.ssh/id_dsa.pub` avec les permissions 644.

Lors de la création, OpenSSH vous demande une *pass phrase* qui est un mot de passe pour protéger la clef privée. Cette *pass phrase* sert à chiffrer la clef privée. La *pass phrase* vous sera alors demandée à chaque utilisation de la clef privée, c'est à dire à chaque fois que vous vous connecterez en utilisant cette méthode d'authentification. Un mécanisme appelé *ssh-agent* permet de ne pas rentrer le mot de passe à chaque fois, comme nous le verrons un peu plus loin dans ce chapitre.

Note

Vous pouvez à tout moment changer la *pass phrase* qui protège votre clef privée avec la commande **ssh-keygen -p**.

Autoriser votre clef publique

Pour cela, il suffit de copier votre clef publique dans le fichier `~/.ssh/authorized_keys` de la machine sur laquelle vous voulez vous connecter à distance. La commande suivante permet de réaliser cette opération via SSH :

```
% ssh-copy-id -i ~/.ssh/id_dsa.pub login@nom_DNS_du_serveur
```

Se connecter

La commande est la même que pour une authentification par mot de passe.

40.5 Transfert de fichiers par SSH

En console

Le transfert de fichiers par SSH est possible d'au moins trois façons :

- avec **scp** (comme Ssh CoPy), qui s'utilise la même manière que la commande **cp** ;
- avec **sftp**, logiciel très basique qui s'utilise comme **ftp**.

– avec **lftp**, dont je vous avais déjà parlé au Chapitre 22 pour les transferts de fichiers par FTP.
  Ici également, vous pouvez utiliser la méthode d'authentification par mot de passe ou par clefs, l'utilisation restant la même dans les deux cas.

Utiliser SCP

Pour illustrer la syntaxe, je vais vous donner quelques exemples :

– pour transférer le fichier `test1.txt` situé dans le répertoire courant vers le home du compte *toto* de la machine *ordi1.exemple.org* sur laquelle tourne un serveur SSH :

```
% scp test1.txt toto@ordi1.exemple.org:
```

– pour récupérer le fichier `test2.txt` situé le home de l'utilisateur *toto* de la machine *ordi2.exemple.org* et l'écrire dans le répertoire courant :

```
% scp toto@ordi2.exemple.org:test2.txt .
```

– pour récupérer tous les fichiers ayant l'extension `.txt` situés dans le répertoire `/usr/local` de la machine *ordi2.exemple.org* et l'écrire dans le sous-répertoire `test-scp` du répertoire courant :

```
% scp toto@ordi2.exemple.org:'/usr/local/*.txt' test-scp
```

– pour transférer l'intégralité du sous-répertoire `test-scp` du répertoire courant vers le sous répertoire `incoming` du home de l'utilisateur *toto* de la machine *ordi1.exemple.org* :

```
% scp -r test-scp toto@ordi1.exemple.org:incoming
```

Utiliser lftp

Je vous avais déjà parlé d'utilisation de *lftp* comme client FTP dans la section 22.2. Mais ce que je ne vous avais pas dit, c'est que *lftp* sait aussi transférer des fichiers par SSH !

Pour l'installation et la configuration de *lftp*, reportez-vous à la section 22.2.

Pour se connecter par SSH en utilisateur *toto* sur le serveur *ordi1.exemple.org* :

```
% lftp sftp://toto@ordi1.exemple.org
```

Ensuite, les commandes sont exactement les mêmes que lors de l'utilisation de lftp comme client FTP !

En graphique

GNOME permet de se connecter à un serveur SSH directement dans Nautilus. Comme pour FTP, cela permet d'accéder aux fichiers distants depuis toutes les applications GNOME.

Pour cela, allez dans le menu *Raccourcis > Se connecter à un serveur*, puis choisissez *SSH*, et réglez les paramètres de connexion (Figure 40.1).

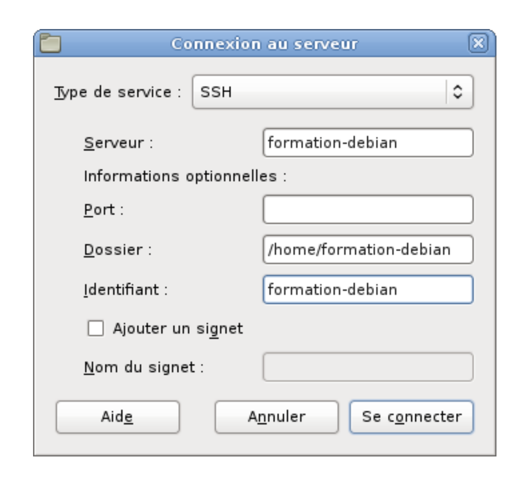


FIGURE 40.1 – Connexion à un serveur SSH

40.6 Se connecter par SSH sans taper de mot de passe

Le principe

Cette section s'adresse à ceux qui utilisent un couple de clefs publiques / privées, et qui ont chiffré leur clé privée avec une *pass phrase* (c'est la configuration la plus sûre). Par conséquent, le client SSH demande la *pass phrase* à chaque utilisation des clefs pour s'authentifier.

Pour éviter d'avoir à taper systématiquement sa *pass phrase*, il faut utiliser *ssh-agent* : ce programme tourne en tâche de fond et garde la clef en mémoire. La commande *ssh-add* permet de donner sa clef à *ssh-agent*. Ensuite, quand vous utilisez le client SSH, il contacte *ssh-agent* pour qu'il lui donne la clef.

La pratique

en console

Dans une console, lancez *ssh-agent* en évaluant les commandes qu'il écrit sur sa sortie :

```
% eval $(ssh-agent)
```

Puis donnez votre clef à l'agent :

```
% ssh-add
```

Il vous demande alors votre *pass phrase*. Maintenant que votre clef a été transmise à l'agent, vous pouvez vous connecter sans entrer de mot de passe à toutes les machines pour lesquelles vous avez mis votre clef publique dans le fichier `~/.ssh/authorized_keys`.

en mode graphique

Si vous utilisez GDM, l'agent SSH a déjà été lancé par GDM. Vous n'avez donc plus qu'à exécuter **ssh-add** une fois que vous êtes connecté.

40.7 Faire des tunnels SSH

Faire un tunnel SSH est un moyen simple de chiffrer n'importe quelle communication TCP entre votre machine et une machine sur laquelle vous avez un accès SSH.

Par exemple, pour établir un tunnel SSH pour une connexion HTTP vers la machine *serveur.exemple.org* :

```
% ssh -L 2012:serveur.exemple.org:80 toto@serveur.exemple.org
```

où *2012* est le port sur la machine cliente à partir duquel la connexion entre dans le tunnel SSH (le port doit être supérieur à 1024 si on ne veut pas avoir à lancer le tunnel en tant que *root*, et le pare-feu ne doit pas bloquer ce port).

Ensuite, il suffit de lancer un navigateur Web en lui demandant de se connecter en local sur ce port :

```
% w3m http://localhost:2012
```

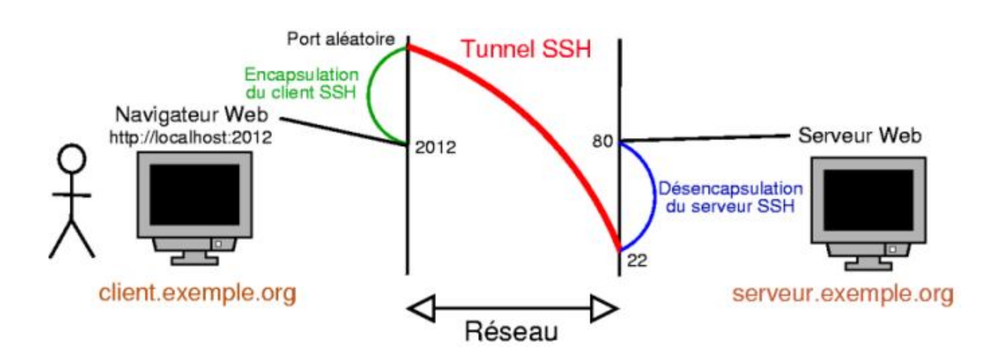


FIGURE 40.2 – Exemple de tunnel SSH

40.8 Et le bon vieux Telnet ?

Qu'est-ce que Telnet ?

Telnet, c'est comme SSH... mais en moins bien ! Telnet est un protocole qui permet d'accéder à distance à une machine, mais la connexion n'est pas sécurisée : le mot de passe et les données sont transférés en clair ! Il est donc conseillé de ne pas utiliser Telnet mais uniquement SSH.

Client et serveur Telnet

Le client Telnet se trouve dans le paquet *telnet*. Ce paquet est installé par défaut.

Le serveur Telnet se trouve dans le paquet *telnetd*. Il n'y a aucune configuration à faire.

Pour se connecter à un serveur Telnet, tapez :

```
% telnet nom_DNS_du_serveur_telnet
```

et ensuite rentrez votre login et votre mot de passe quand il vous le demande.

Chapitre 41

Transférer des fichiers

Il existe deux façons de servir des fichiers :

le transfert de fichiers qui permet de télécharger et de téléverser des fichiers sur un serveur : c'est l'objet de ce chapitre ;

le partage de fichiers qui permet d'accéder à des fichiers et à des répertoires depuis des ordinateurs distants comme s'ils étaient présents sur un périphérique local : techniquement, il s'agit alors de *systèmes de fichiers réseau*, qui feront l'objet du Chapitre 43 et du Chapitre 44.

Traditionnellement, on utilise pour cela le protocole FTP (*file transfer protocol*). Ce protocole souffre cependant de grands défauts, et est largement déprécié au profit des protocoles HTTP (*hypertext transfer protocol*, le protocole du Web) et SFTP (*SSH file transfer protocol*). Ce sont donc ces deux systèmes que nous vous proposons de découvrir.

41.1 Téléchargement anonyme : HTTP

Installation du serveur d'Apache HTTPD

Si vous souhaitez mettre à disposition des fichiers, sur l'Internet ou sur votre réseau local, le plus simple est d'installer un serveur HTTP comme Apache HTTPD :

```
# aptitude install apache2
```

Après son installation, le serveur HTTP démarre. Par défaut, il est configuré pour servir les fichiers contenus dans le répertoire `/var/www/`. Ce répertoire contient déjà un fichier HTML de démonstration, `index.html`.

Mettre à disposition des fichiers

Pour proposer des fichiers en téléchargement, vous pouvez tout simplement les placer dans ce répertoire `/var/www/` :

```
# cp formation-debian.pdf /var/www/
```

Tous les fichiers que vous placez dans ce répertoire seront accessibles à l'URL `http://nom_du_serveur/nom_du_fichier`, dans notre cas : `http://formation-debian.via.ecp.fr/formation-debian.pdf`.

Proposer une liste des fichiers

Pour le moment, lorsque quelqu'un accède à l'URL racine de votre serveur, `http://nom_du_serveur/`, c'est le fichier de démonstrations `/var/www/index.html` qui s'affiche. Apache est capable d'afficher à la place une liste des fichiers disponibles : il suffit pour cela de retirer le fichier `index.html` :

```
# rm /var/www/index.html
```

Espace personnel par utilisateur

Si vous êtes seul à utiliser votre ordinateur, vous pouvez sans problème utiliser le répertoire `/var/www/` pour diffuser vos fichiers, quoiqu'il faille à chaque fois effectuer les copies ou déplacements en tant que root.

Il est également possible de proposer à chaque utilisateur la possibilité d'avoir un espace Web personnel. Pour cela, il faut activer un module du serveur Apache HTTPD, puis redémarrer celui-ci :

```
# a2enmod userdir
Enabling module userdir.
Run '/etc/init.d/apache2 restart' to activate new ↩
    configuration!
# /etc/init.d/apache2 restart
* Restarting web server apache2
 ... waiting                                          [  ↩
     OK ]
```

Après cela, un utilisateur *ksaffier* peut créer dans son répertoire personnel un sous-répertoire nommé `public_html` : tout ce qu'il y placera sera alors disponible à l'URL `http://nom_du_serveur/~ksaffier` :

```
% mkdir ~/public_html
% cp formation-debian.pdf ~/public_html/
```

ASTUCE

Vous pouvez également construire dans ce répertoire un site Web personnel !

41.2 Téléchargement identifié et sécurisé : SFTP

Un serveur HTTP permet de mettre à disposition des fichiers, qui peuvent alors être téléchargés par n'importe qui. En revanche, cela ne peut pas convenir si vous souhaitez contrôler l'accès à ces fichiers, ou si vous souhaitez déposer des fichiers sur votre serveur. Pour cela, la meilleure solution consiste à utiliser le protocole SFTP.

Installation

SFTP signifie *SSH file transfer protocol* : il s'agit donc d'un protocole lié à SSH, et qui est intégré au serveur OpenSSH. Ce serveur a fait l'objet du Chapitre 40, auquel je vous renvoie si vous ne l'avez pas déjà installé.

Charger et déposer des fichiers

Comme expliqué au Chapitre 40, vous pouvez accéder à votre serveur SFTP avec :

avec la commande sftp utilisateur@serveur assez limitée, son utilisation, très basique est semblable à celle du programme **ftp**, avec des commandes comme **get** et **put** ;

avec la commande lftp sftp://utilisateur@serveur beaucoup plus agréable, ce logiciel est décrit au Chapitre 22 ;

sous GNOME directement avec Nautilus, le gestionnaire de fichiers de GNOME ;

sous Windows , avec un logiciel comme WinSCP.

Limiter l'accès au répertoire personnel

Il est possible de limiter l'accès de certains utilisateurs, pour qu'ils puissent par exemple administrer leur site Web personnel, mais n'aient pas le droit d'accéder aux fichiers des autres utilisateur, ni à quoi que ce soit d'autre d'ailleurs.

Il est probable que vous ne souhaitiez pas limiter ainsi tous vos utilisateurs, parce que vous n'auriez alors plus la possibilité de vous connecter vous-même à votre serveur par SSH. Nous allons donc créer un groupe qui servira à identifier les utilisateurs qui doivent être ainsi limitées, et y ajouter ces utilisateurs :

```
# addgroup chrooted
# adduser ksaffier chrooted
```

Nous allons maintenant éditer le fichier de configuration du serveur OpenSSH, `/etc/ssh/sshd_config`. Tout d'abord, il faut modifier ainsi la ligne **Subsystem sftp** :

```
Subsystem sftp internal-sftp
```

Ajoutez ensuite, *à la fin du fichier*, la section qui permettra de contraindre tous les membres du groupe précédemment défini dans leur répertoire personnel, et de ne les autoriser à faire que du SFTP :

```
Match Group chrooted
    ChrootDirectory %h
    ForceCommand internal-sftp
```

Explications : la première ligne demande de n'appliquer la suite qu'aux utilisateurs membres du groupe *chrooted*. La seconde définit un répertoire « prison » ou *chroot*, le **%h** désignant le répertoire personnel de l'utilisateur. Enfin, la dernière ligne force l'utilisation de SFTP, ce qui interdit par exemple les connexions SSH normales.

AVERTISSEMENT

Cette dernière ligne est importante. En effet, si vous l'omettez, vos utilisateurs ne pourront pas, dans un premier temps, ouvrir de connexion SSH, puisqu'ils sont contraints de rester dans leur répertoire personnel, qui ne contient pas de Shell. Or, ils pourront tout à fait déposer eux-mêmes un Shell dans leur répertoire et pourront alors se connecter normalement ! Ils resteront toujours limités dans leur *chroot*, mais comme il existe des moyens de sortir d'un *chroot*, autant éviter cette éventualité, et ne fournir d'accès SSH qu'aux utilisateurs auxquels vous êtes sûr de pouvoir faire confiance.

Dernière opération : le répertoire de *chroot* doit appartenir à l'utilisateur *root*. Il va donc falloir changer le propriétaire du répertoire personnel de chaque utilisateur ainsi limité, mais également leur donner un moyen de poser des fichiers ! Je vous propose donc de leur créer un sous-répertoire `public_html/`, dans lequel ils pourront contruire leur site Web personnel :

```
# chown root:root ~ksaffier
# mkdir ~ksaffier/public_html
# chown ksaffier ~ksaffier/public_html
```

Lorsque vos utilisateurs se connectent en SFTP, ils sont donc placés dans leur répertoire personnel comme racine virtuelle. Si vous souhaitez que vos utilisateurs arrivent directement dans le répertoire où ils doivent déposer leurs documents (`/public_html/`, de leur point de vue), vous pouvez leur créer un pseudo-répertoire personnel pointant vers ce répertoire. En effet, OpenSSH essaie de passer dans un répertoire personnel *à l'intérieur de la racine virtuelle* :

```
    # mkdir ~ksaffier/home
    # ln -s ../public_html ~ksaffier/home/ksaffier
```

C'est fait : maintenant, l'utilisateur *ksaffier* peut se connecter en SFTP à votre serveur, mais ne peut accéder qu'à son répertoire personnel, et ne peut écrire que dans son répertoire `public_html/`.

Chapitre 42

Faire du déport d'affichage

42.1 Qu'est-ce que le déport d'affichage ?

Le déport d'affichage ou *export display* consiste à se logguer à distance en mode graphique, comme on le fait avec un client et un serveur SSH en mode texte. On peut alors exécuter des applications graphiques sur le serveur distant : la fenêtre graphique de l'application et son contenu seront envoyés par le réseau vers la machine cliente ; les données du clavier et de la souris de la machine cliente sont envoyées vers le serveur.

Note
L'export display nécessite une bonne connexion réseau entre le client et le serveur puisque le serveur envoie des images de l'écran au client.

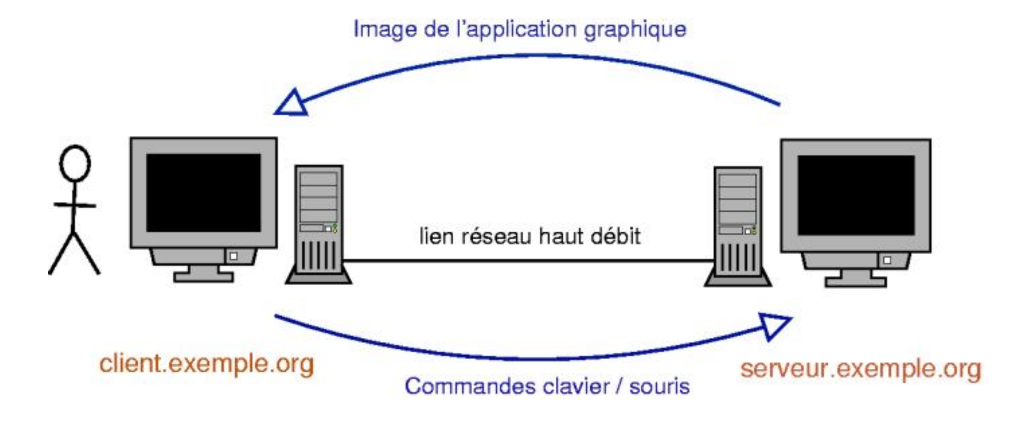


FIGURE 42.1 – Le principe de l'export display

42.2 Se connecter à un Unix/Linux à distance

Depuis un système Unix/Linux

Il y a plusieurs possibilitées de connexion, sachant que seule la première est entièrement chiffrée.

Note

Dans le modèle du protocole X11, qui est utilisé sous Unix, le *serveur graphique* désigne la machine qui reçoit les données du clavier et de la souris, et envoie des images à une carte graphique, ou plus précisément le logiciel qui gère ces interaction, X.org. Les *clients graphiques* désignent les logiciels qui utilisent ces fonctions d'entrée et d'affichage.

Dans le cas du déport d'affichage, les clients graphiques sont exécutés sur une machine distante, et affichés sur le serveur graphique de votre ordinateur local. Pour cela, il n'est pas nécessaire d'installer un serveur graphique sur l'ordinateur distant !

Déport d'affichage par SSH

SSH possède une fonction de déport d'affichage. Il faut que le serveur SSH distant ait autorisé cette fonction, comme expliqué au Chapitre 40. Pour l'utiliser, tapez dans un *xterm* :

```
% ssh -X login@serveur.exemple.org
```

puis lancez l'application graphique de votre choix depuis le *shell* que vous obtenez sur le serveur.

Déport d'affichage avec XDMCP

Le protocole XDMCP permet de se connecter au serveur graphique d'une machine distante, et de faire exactement comme si on était connecté sous X en local sur la machine.

AVERTISSEMENT

Ce protocole fait tout transiter en clair sur le réseau, y compris les mots de passe !

Pour cela, *GDM* doit être installé sur le serveur. Toujours sur le serveur, éditez le fichier `/etc/gdm/gdm.conf` et modifiez la section *xdmcp* :

```
[xdmcp]
Enable=true
```

Redémarrez GDM :

```
# /etc/init.d/gdm restart

Stopping GNOME Display Manager: gdm.
Starting GNOME Display Manager: gdm.
```

Sur le client, vous pouvez alors vous connecter au serveur graphique distant. Depuis le GDM du client, allez dans le menu : *Actions* > *Exécuter le sélecteur XDMCP*, puis rentrez le nom ou l'IP de votre serveur XDMCP. vous devriez alors arriver sur la fenêtre GDM du serveur.

Depuis un système Windows

Il suffit d'installer sur la machine Windows un serveur X, comme XMing, ainsi qu'un client SSH comme PuTTY.

Son installation et son utilisation, notamment pour faire de l'export display de GNU/Linux vers Windows, sont expliquées dans l'annexe Annexe G.

42.3 Se connecter à distance à un Windows depuis un Linux

Il y a plusieurs possibilités, dont seule la seconde utilise des connexions chiffrées :

– par VNC vers un Windows sur lequel tourne un serveur VNC : installez le paquet **xvncviewer** qui contient un client VNC puis lancez la commande suivante pour vous connecter au serveur VNC :

```
% xvncviewer serveur.exemple.org
```

– par Terminal Serveur vers un Windows 2000 Server ou Remote Desktop vers un Windows XP Pro (le protocole est le même) : installez le paquet **rdesktop** qui contient un client RDP (Remote Desktop Protocol) et lancez la commande suivante pour vous connecter au Windows distant :

```
% rdesktop -u login serveur.exemple.org
```

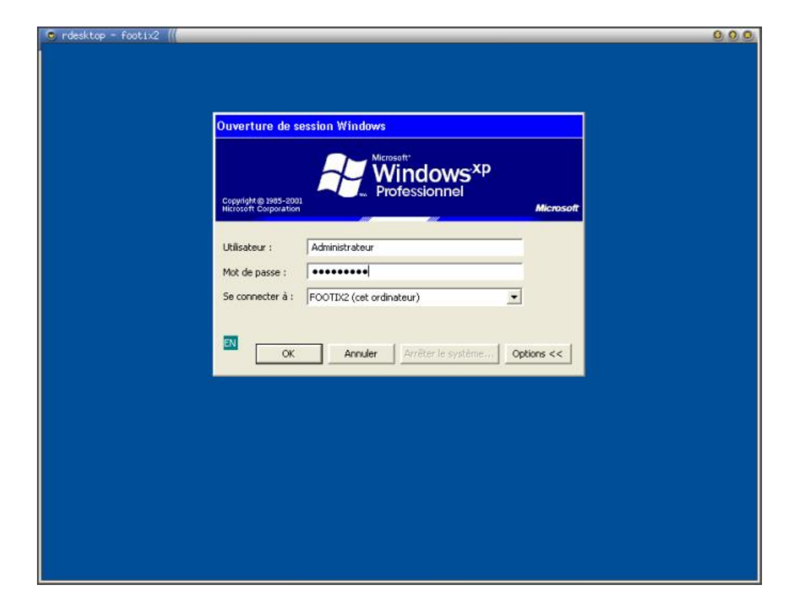


FIGURE 42.2 – rdesktop

Chapitre 43

NFS : le partage de fichiers sous Unix

NFS (Network File System) est un protocole standard de partage de répertoires sous Unix/Linux. Dans ce chapitre, nous allons vous apprendre à partager un répertoire par NFS, puis à le monter sur un système client pour pouvoir l'utiliser.

43.1 NFS côté serveur

Configuration nécessaire

Il faut installer le paquet *nfs-kernel-server* :

```
# aptitude install nfs-kernel-server
```

Partager un répertoire

Editez le fichier `/etc/exports` et rajoutez la ligne suivante pour partager le répertoire `/home/test/` à la machine *ordi2.exemple.org* :

```
/home/test  ordi2.exemple.org(rw,root_squash)
```

L'option **rw** permet d'exporter en *lecture-écriture* (utiliser **ro** pour exporter en *lecture seule*). L'option **root_squash** spécifie que le root de la machine *ordi2.exemple.org* n'a pas les droits de root sur le répertoire partagé (l'option **no_root_squash** spécifie que le root de la machine sur laquelle le répertoire est monté a les droits de root sur le répertoire). L'option *root_squash* est l'option par défaut.

Note

L'option **rw** signifie en réalité que l'utilisateur dont l'ID est 1001 (par exemple...) sur le client NFS a les droits d'écriture sur les fichiers et les répertoires qui appartiennent à l'utilisateur dont l'ID est 1001 sur le serveur NFS. Attention, ces utilisateurs n'ont pas forcément le même nom de compte Unix et ne correspondent pas forcément aux mêmes personnes !

Enfin, demandez à *nfs-kernel-server* de relire sa configuration :

```
# /etc/init.d/nfs-kernel-server reload
* Re-exporting directories for NFS kernel daemon... [ OK ]
```

43.2 NFS côté client

Pour monter le répertoire `/home/ftp/` partagé par la machine dont le nom DNS est *ordi1.exemple.org* dans le répertoire `/mnt/test` déjà crée, utilisez la commande **mount** :

```
# mount -t nfs ordi1.exemple.org:/home/ftp /media/test
```

Une fois que vous n'avez plus besoin de ce partage, vous pouvez le démonter :

```
# umount /media/test
```

Pour que ce répertoire soit monté à chaque démarrage, rajoutez la ligne suivante dans le fichier de configuration `/etc/fstab` :

```
ordi1.exemple.org:/home/ftp  /media/test   nfs    soft,timeo ↩
    =5,intr,rsize=8192,wsize=8192  0  0
```

Pour comprendre les options, regardez leur description dans **man mount**.

Chapitre 44

Le voisinage réseau Windows sous Linux

L'objectif de ce chapitre est d'être capable de surfer sur le voisinage réseau Windows (c'est l'aspect client), et de pouvoir mettre en partage des fichiers (c'est l'aspect serveur). Pour cela, nous allons utiliser les outils développés par le projet GPL Samba.

Note

Le « voisinage réseau » utilise le protocole SMB (Server Message Block), créé à l'origine par IBM, puis repris par Microsoft. Par souci d'ouverture, ceux-ci ont renommé ce protocole en CIFS (Common Internet FileSystem). Lors de la publication de Windows Vista, Microsoft a repris la dénomination SMB pour sa nouvelle version SMB 2. SMB et CIFS désignent donc, pour l'essentiel, le même protocole.

44.1 Samba côté serveur

Installation des paquets

Il suffit d'installer le paquet *samba*, qui va installer par dépendance le paquet *samba-common* qui sert à la fois au serveur *et* au client :

```
# aptitude install samba
```

Configuration

Dès l'installation, il vous pose quelques question, auxquelles vous pouvez répondre en laissant les propositions pré-remplies. En effet, nous vous proposons de télécharger un fichier de configuration que vous allez personnaliser.

Le fichier de configuration de Samba est `/etc/samba/smb.conf`. Ce fichier a une petite partie commune au client et au serveur, le reste étant spécifique au serveur. Copiez mon fichier de configuration d'exemple dans le répertoire `/etc/samba/` :

```
# cp ~/fichiers-config/smb.conf /etc/samba/
```

ou :

```
% wget http://formation-debian.via.ecp.fr/fichiers-config/smb ↩
   .conf
# mv smb.conf /etc/samba/
```

Editez le fichier et personnalisez les paramètres en lisant les instructions en commentaire. Une fois que le fichier est bien personnalisé, vérifiez s'il est bien valide avec le programme **testparm** :

```
# testparm
```

Si le fichier de configuration est bien valide, vous pouvez alors demander à Samba de relire son fichier de configuration :

```
# /etc/init.d/samba reload
Reloading /etc/samba/smb.conf (smbd only).
```

Note

Il n'y a pas de processus *samba* qui tourne en tâche de fond pour le serveur, mais deux processus : *smbd* qui s'occupe des partages et de l'authentification, et *nmbd* qui s'occupe de la résolution des noms NetBIOS.

44.2 Samba côté client

En console

Installation du client en mode texte

Le client pour le voisinage réseau Windows en console s'appelle *smbclient* :

```
# aptitude install smbclient
```

Utiliser smbclient

Voici une liste des commandes les plus utiles (à chaque fois il vous demandera de taper votre mot de passe pour le domaine Windows) :

– Pour avoir la liste des machines du voisinage réseau :

```
% smbclient -L ↩
   nom_DNS_ou_NetBIOS_du_contrôleur_de_domaine -U ↩
   login_Windows
```

– Pour voir les partages de l'ordinateur *ordi1* :

```
% smbclient -L ordi1 -U login_Windows
```

– Pour télécharger le fichier *fichier1.txt* disponible dans le partage *partage1* de l'ordinateur *ordi1* :

```
% smbclient //ordi1/partage1 -U login_Windows
smb: \> ls
smb: \> cd répertoire_distant
smb: \> lcd répertoire_local
smb: \> get fichier1.txt
smb: \> quit
```

– Pour imprimer le fichier PostScript *fichier2.ps* sur l'imprimante *printer2* partagée par l'ordinateur *ordi2* :

```
% smbclient //ordi2/printer2 -U login_Windows
smb: \> print fichier2.ps
smb: \> quit
```

– Pour déposer le fichier *fichier3.txt* sur le partage *incoming* partagée en écriture par l'ordinateur *ordi3* :

```
% smbclient //ordi3/incoming -U login_Windows
smb: \> put fichier3.txt
smb: \> quit
```

– Pour changer son mot de passe du voisinage réseau Windows :

```
% smbpasswd -r ↩
    nom_DNS_ou_NetBIOS_du_contrôleur_de_domaine -U ↩
    login_Windows
```

Monter un partage du voisinage réseau

Il faut installer le paquet *smbfs* :

```
# aptitude install smbfs
```

Pour monter le partage *partage1* de l'ordinateur *ordi1* dans le répertoire `/mnt/test/` déjà existant :

```
# mount -t cifs //ordi1/partage1 /mnt/test -o username= ↩
    login_Windows
```

Quand vous n'en n'avez plus besoin, vous pouvez démonter le système de fichiers :

```
# umount /mnt/test
```

En mode graphique

GNOME permet de se connecter à un serveur SMB/CIFS directement depuis Nautilus. Comme pour le FTP, cela permet d'accéder aux fichiers distants depuis toutes les applications du bureau GNOME.

Pour cela, allez dans le menu *Raccourcis* > *Se connecter à un serveur*, puis choisissez *Partage Windows* et réglez les paramètres de connexion (Figure 44.1).

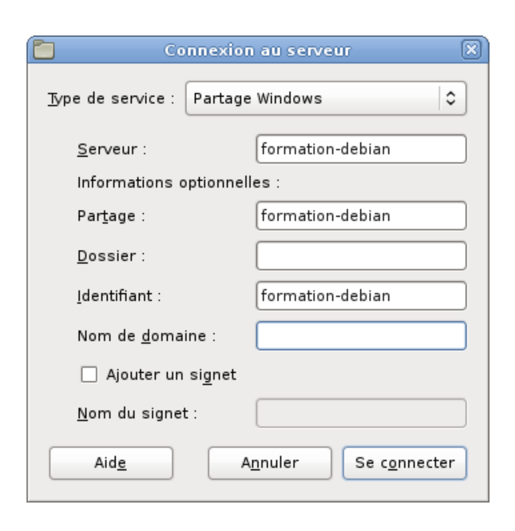


FIGURE 44.1 – Connexion à un serveur SMB/CIFS

Chapitre 45

Se synchroniser sur un serveur de temps

45.1 Qu'est-ce que le protocole NTP ?

NTP signifie *Network Time Protocol*. C'est un protocole qui permet à un ordinateur de synchroniser son horloge sur un autre ordinateur de précision plus élevée. Il existe ainsi des serveurs NTP de différents niveaux (appelés *strates*) qui correspondent à différentes précisions. Les horloges atomiques forment la strate 0, et sont directement reliées aux serveurs de strate 1, qui ne sont eux-même accessibles qu'à des serveurs de strates 2 ou 3, certains de ces serveurs étant librement accessibles.

Nous vous proposons dans ce chapitre de vous synchroniser sur le réseau public des serveurs de temps de votre pays.

45.2 Se synchroniser sur le réseau NTP

Installation d'un client NTP

Installez le programme *ntp* :

```
# aptitude install ntp
```

Configuration

ntp utilise plusieurs serveurs pour se mettre à l'heure, afin d'augmenter sa précision en cas d'indisponibilité ou d'erreurs d'un serveur. Leurs adresses sont écrites dans le fichier `/etc/ntp.conf`, dans les différents champs *server*. Pour le moment, Debian y a placé une liste de serveurs répartis dans le monde, mais il est plus efficace d'utiliser des serveurs proches, situés dans le même pays. Si vous êtes en France, vous pouvez donc remplacer ces lignes par :

```
server 0.fr.pool.ntp.org iburst dynamic
server 1.fr.pool.ntp.org iburst dynamic
server 2.fr.pool.ntp.org iburst dynamic
server 3.fr.pool.ntp.org iburst dynamic
```

L'option *iburst* signifie qu'en cas d'indisponibilité du serveur, **ntp** essaiera plusieurs fois avant d'abandonner. L'option *dynamic* permet de conserver dans la configuration les serveurs indisponibles, au cas où ils seraient à nouveau accessibles plus tard.

ntp est en fait un *démon*, qui tourne en permanence sur votre système afin de maintenir l'horloge à l'heure et de corriger sa dérive. Après avoir modifié sa configuration, vous devez donc le relancer :

```
# /etc/init.d/ntp restart
 * Stopping NTP server ntpd [ OK ]
 * Starting NTP server ntpd [ OK ]
```

Cinquième partie

Debian GNU/Linux en console

Une des originalités de GNU/Linux est d'avoir deux modes : le mode console et le mode graphique. Si le mode graphique est plus familier pour les utilisateurs venant du monde Windows, je vais essayer dans ce chapitre de vous faire découvrir le mode console et ses avantages.

Le principal avantage du mode console est de pouvoir accéder facilement à ses applications en mode console depuis n'importe quel ordinateur connecté à Internet quel que soit son système d'exploitation et le débit de sa connexion Internet. Cela est permis par l'utilisation de 40 et éventuellement de *screen*, qui est expliqué à la fin de cette partie. En effet, il existe des clients SSH pour tous les systèmes d'exploitation (cf Annexe G pour avoir un exemple de client SSH Windows), et vous pouvez vous connecter par SSH depuis n'importe quel ordinateur connecté à Internet vers un serveur sur lequel vous avez un compte et ensuite lancer vos applications en console préférées.

```
e:Quitter  -:PgPréc  <Space>:PgSuiv v:Voir attach.  d:Effacer  r:Répondre  j:Suivant ?:Aide
  56     sep 15 svn-formation-d (   0) SVN commit r688 - in trunk/src: . annexes part1 part
  57     sep 22 svn-formation-d (   0) SVN commit r689 - in trunk/src: . fichiers part3
  58     sep 27 svn-formation-d (   0) SVN commit r690 - in trunk/src: annexes fichiers par
  59     sep 27 svn-formation-d (   0) SVN commit r691 - in trunk/src: . annexes fichiers p
  60     sep 27 svn-formation-d (   0) SVN commit r692 - in trunk: fichiers-config src/fich
--Mutt: =formation-debian-devel [Msgs:60 1,0M]--(threads/date)----------------(end)--
Date: Sat, 27 Sep 2008 05:02:02 +0200 (CEST)
From: svn-formation-devel@via.ecp.fr
To: formation-debian-devel@via.ecp.fr
Subject: SVN commit r691 - in trunk/src: . annexes fichiers part3 part4 unused
Reply-to: formation-debian-devel@via.ecp.fr
X-list: formation-debian-devel

Author: tanguy
Date: Sat Sep 27 05:02:01 2008
New Revision: 691

Log:
Mise à jour de la partie 4 (réseau) finie.

Qu'est-ce que le chapitre sur Subversion faisait dans ce chapitre ? Je l'ai mis dans les
annexes, mais on peut l'en retirer puisqu'il est dans le wiki. Retiré le chapitre sur CVS,
déprécié, plus personne ne l'utilise.

-   - 59/60: svn-formation-devel@   SVN commit r691 - in trunk/src: . annexes fichi -- (4%)
```

FIGURE 45.1 – Exemple d'application en console

Note

Si vous avez déjà travaillé avec le mode MS-DOS de Windows, vous vous êtes sans doute rendu compte qu'il s'agissait d'une interface extrêmement austère et peu pratique. Ne vous inquiétez pas, comme vous avez pu le voir au début de cette formation, le mode texte de GNU/Linux et des systèmes Unix en général est largement plus puissant et plus agréable à utiliser !

Chapitre 46

Le courrier en console

Je vous propose la solution suivante si vous voulez lire votre courrier en console :

- utiliser fetchmail pour relever votre courrier sur votre ordinateur (dans le cas où l'ordinateur sur lequel vous comptez lire votre courrier n'est pas le même que celui qui héberge votre boîte aux lettres) ;
- utiliser Procmail pour le trier et SpamAssassin pour filtrer le spam ;
- utiliser Mutt comme lecteur de courrier.

46.1 Relever son courrier

Commencez par installer fetchmail :

```
# aptitude install fetchmail
```

Écrivez votre fichier de configuration `.fetchmailrc` dans votre répertoire personnel en vous basant sur mon fichier de configuration type :

```
% cp /root/config/fetchmailrc ~/.fetchmailrc
```

ou :

```
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
   fetchmailrc
% mv fetchmailrc ~/.fetchmailrc
```

et modifiez le nom du serveur POP, le login et le mot de passe.

Le fichier de configuration doit avoir les permissions 600, sinon fetchmail refuse de fonctionner :

```
% chmod 600 ~/.fetchmailrc
```

Pour rapatrier les messages depuis le serveur sur votre ordinateur, il suffira de lancer la commande :

```
% fetchmail
```

mais avant ça, nous allons mettre en place le filtrage avec Procmail et SpamAssassin.

46.2 Trier son courrier avec Procmail et SpamAssassin

Le principe

Procmail est un programme simple capable de trier les messages en regardant leurs en-têtes. Nous allons lui demander de trier le courrier dans différents répertoires, appelés *mailbox* (boîtes aux lettres).

SpamAssassin est un programme qui parcourt les en-têtes, le message et les éventuelles pièces jointes et les analyse. Pendant cette analyse, il regarde un certain nombre de critères généralement vérifiés par les spams, à chaque critère étant associé un nombre de points (plus le critère est significatif, plus le nombre de points est grand). Enfin, il additionne les points pour attribuer au message une note globale. Par défaut, si la note globale atteint 5.0 (on peut la changer en modifiant son fichier de configuration `~/.spamassassin/user_prefs` qui est créé automatiquement à la première utilisation), le message est considéré comme du spam. *Spamassassin* rajoute alors un en-tête *X-Spam-Status: Yes* et un rapport de l'analyse dans le corps du message. L'en-tête rajouté par *spamassassin* permet par exemple à *procmail* d'aiguiller le spam dans une boîte aux lettres poubelle.

Commencez par installer ces deux programmes (Procmail est normalement déjà installé) :

```
# aptitude install procmail spamassassin
```

Le principe du tri de courrier avec Procmail et SpamAssassin est le suivant :

1. le courrier reçu par Postfix via fetchmail arrive dans le fichier `/var/mail/nom_de_l'utilisateur`,
2. Postfix lance Procmail, qui regarde si l'utilisateur a un fichier `.procmailrc` dans son répertoire personnel,
3. si oui, il lit ce fichier `~/.procmailrc` et suit les instructions qu'il contient,
4. dans notre cas, ces instructions lui disent de déplacer le courrier dans une des boîtes aux lettres contenues dans le répertoire `~/mail`. SpamAssassin est alors éventuellement appelé par Procmail pour détecter les messages indésirables, qui sont ensuite envoyés dans la boîte aux lettres `spam`.

AVERTISSEMENT

Attention, une mauvaise utilisation de Procmail peut aboutir à la perte de messages. Il faut donc toujours faire des tests après la mise en place et après chaque modification de sa configuration.

La pratique

Utilisez le fichier de configuration d'exemple pour Procmail `procmailrc` :

```
% cp /root/config/procmailrc ~/.procmailrc
```

ou :

```
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
  procmailrc
% mv procmailrc ~/.procmailrc
```

Regardez les commentaires écrits dans ce fichier pour comprendre comment il est construit et personnalisez-le. Aidez-vous éventuellement des manuels **man procmailrc** et **man procmailex**.

Créez un sous-répertoire `mail` dans votre répertoire personnel qui accueillera toutes vos boîtes aux lettres :

```
% mkdir ~/mail
```

Les boîtes aux lettres seront des sous-répertoires du répertoire `mail` et seront crées automatiquement par Procmail au premier message aiguillé dedans.

46.3 Lire son courrier avec Mutt

Installer et configurer Mutt

Le paquet *mutt* est normalement déjà installé.

Le fichier de configuration de Mutt est `~/.muttrc`. Utilisez mon fichier de configuration d'exemple comme base de travail :

```
% cp /root/config/muttrc ~/.muttrc
```

ou :

```
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
    muttrc
% mv muttrc ~/.muttrc
```

Éditez-le et personnalisez au minimum les deux premiers paramètres (le champ From et la liste des adresses électroniques utilisées). Avec le fichier de configuration que je vous propose, Mutt détermine la liste des boîtes aux lettres à partir de la liste des sous-répertoires du répertoire `~/mail/` (les sous-répertoires correspondant aux boîtes aux lettres sont créés à la réception du premier messages pour cette boîte aux lettres). Pour avoir la liste complète des paramètres et de leur description, utilisez **man muttrc** ou ce manuel d'utilisation en français.

Si vous le désirez, vous pouvez écrire un carnet d'adresse pour Mutt, sous forme d'un fichier texte `~/.aliases`. Chaque ligne du fichier correspond à une entrée dans le carnet d'adresse. À chaque entrée dans le carnet d'adresse, vous devez associer une clef que vous pourrez alors utiliser dans *mutt* à la place du nom et de l'adresse électronique complète. Les lignes de fichier doivent avoir la syntaxe suivante :

```
alias clef Nom_et_Prénom <adresse_électronique>
```

Par exemple, si vous voulez me mettre dans votre carnet d'adresse en me désignant par la clef *alex*, ajoutez la ligne :

```
alias alex Alexis de Lattre <alexis@via.ecp.fr>
```

Utiliser Mutt

Commencez par lancer Mutt :

```
% mutt -y
```

La liste des boîtes aux lettres apparaît à l'écran : Figure 46.1.

```
y:Quitter  c:Changement de répertoire  m:Masque  ?:Aide
 1      drwx------  5 tanguy   tanguy       4,0K sep 27 14:51 =formation-debian-devel/
 2      drwx------  5 tanguy   tanguy       4,0K sep 27 14:51 =formation-debian-users/
 3      drwx------  5 tanguy   tanguy       4,0K sep 27 14:51 =inbox/

-- Mutt: Boîtes aux lettres [0]
```

FIGURE 46.1 – Mutt : liste des boîtes aux lettres

Sélectionnez une boîte aux lettres pour lire les messages qui s'y trouvent (Figure 46.2). Ensuite, pour passer d'une boîte aux lettres à une autre, utilisez la touche **c** : si une boîte aux lettres contient un nouveau message, son nom apparaît et vous n'avez plus qu'à taper **Entrée** ; si vous voulez aller à une boîte aux lettres particulière, tapez son nom précédé du signe **=** (la complétion automatique marche).

```
e:Quitter  -:PgPréc  <Space>:PgSuiv v:Voir attach.  d:Effacer  r:Répondre  j:Suivant ?:Aide
  56     sep 15 svn-formation-d (   0) SVN commit r688 - in trunk/src: . annexes part1 part
  57     sep 22 svn-formation-d (   0) SVN commit r689 - in trunk/src: . fichiers part3
  58     sep 27 svn-formation-d (   0) SVN commit r690 - in trunk/src: annexes fichiers par
  59     sep 27 svn-formation-d (   0) SVN commit r691 - in trunk/src: . annexes fichiers p
  60     sep 27 svn-formation-d (   0) SVN commit r692 - in trunk: fichiers-config src/fich
---Mutt: =formation-debian-devel [Msgs:60 1,0M]---(threads/date)--------------(end)---
Date: Sat, 27 Sep 2008 05:02:02 +0200 (CEST)
From: svn-formation-devel@via.ecp.fr
To: formation-debian-devel@via.ecp.fr
Subject: SVN commit r691 - in trunk/src: . annexes fichiers part3 part4 unused
Reply-to: formation-debian-devel@via.ecp.fr
X-list: formation-debian-devel

Author: tanguy
Date: Sat Sep 27 05:02:01 2008
New Revision: 691

Log:
Mise à jour de la partie 4 (réseau) finie.

Qu'est-ce que le chapitre sur Subversion faisait dans ce chapitre ? Je l'ai mis dans les
annexes, mais on peut l'en retirer puisqu'il est dans le wiki. Retiré le chapitre sur CVS,
déprécié, plus personne ne l'utilise.

-   - 59/60: svn-formation-devel@   SVN commit r691 - in trunk/src: . annexes fichi -- (4%)
```

FIGURE 46.2 – Mutt : messages dans une boîte aux lettres

Pour écrire une lettre, placez-vous dans n'importe quelle boîte aux lettres et tapez **m**. Saisissez l'adresse du destinataire ou sa clef si vous l'avez mis dans votre fichier d'alias, puis le sujet du message. Mutt ouvre alors votre éditeur de texte préféré (Vim si vous n'avez pas changé ce paramètre dans le fichier de configuration). Le texte contient déjà des en-têtes que vous pouvez compléter ou modifier. Tapez le texte du message en dessous des en-têtes :Figure 46.3

```
From: Tanguy Ortolo <tanguy@ortolo.eu>
To: Formation Debian <formation-debian@via.ecp.fr>
Cc:
Bcc:
Subject: Publication de la formation Debian GNU/Linux 5.0 Lenny !
Reply-To:

Salut les debianistes !

J'ai le plaisir de vous annoncer la publication d'une nouvelle édition
de notre formation Debian GNU/Linux, consacrée à la dernière version de
Debian : Lenny, numérotée 5.0 !

Librement,

--
Tanguy Ortolo, mainteneur
~
/tmp/mutt-glamdring-1000-9592-5[+]                        14,11 Tout
-- INSERTION --
```

FIGURE 46.3 – Mutt : rédaction d'un message

Une fois que vous avez fini, enregistrez et quittez. Vous arriver alors dans l'écran d'envoi des messages, dans lequel vous pouvez utiliser les commandes suivantes :

- **t** pour modifier la liste des destinataires,
- **c** pour modifier la liste des personnes en copie (*Cc*),
- **b** pour modifier la liste des personnes en copie cachée (*Bcc*),
- **s** pour modifier le *Subject*,
- **e** pour rééditer le message,
- **a** pour attacher des fichiers au message,
- **p** pour accéder aux fonctions de chiffrement et de signature de *GnuPG* (voir ci-dessous),
- **q** pour annuler l'envoi,
- **y** pour envoyer le message.

Dans une boîte aux lettres, que vous soyez dans la liste des messages ou en train d'afficher un message, vous avez accès aux touches suivantes :

- **r** pour répondre à l'expéditeur du message sélectionné,
- **g** pour répondre à l'expéditeur et aux autres destinataires du message sélectionné,
- **Tab** pour aller au prochain message non lu,
- **v** pour afficher la liste des fichiers attachés à un message puis **s** pour les enregistrer sur le disque dur,
- **h** pour afficher le message avec tous ses en-têtes (et de nouveau **h** pour enlever l'affichage des en-têtes),
- **f** pour transférer le message sélectionné à quelqu'un,
- **d** pour marquer le message comme devant être effacé,
- **u** pour annuler le marquage d'effacement,
- **F** pour marquer le message comme important, i.e. afficher un point d'exclamation au niveau du message (idem pour enlever le marquage),
- **a** pour ajouter ou éditer l'entrée de l'expéditeur du message dans le carnet d'adresse,
- **$** pour réactualiser l'affichage de la boîte aux lettres (effacer les messages marqués comme devant être effacés et afficher les messages qui viennent d'arriver),
- **G** pour aller directement à la liste des boîtes aux lettres.

Quand vous êtes dans la liste des messages d'une boîte aux lettres (et non quand vous êtes entrain d'afficher un message), vous avez en plus accès aux touches suiv-

antes :
- **o** pour trier les messages de la boîte aux lettres : il propose alors le choix entre un tri par date, par expéditeur, par destinataire, par sujet, par discussion (tri normal), etc.,
- **w** pour mettre un marqueur sur un message : il propose alors une liste des marqueurs (cela permet notamment de remettre un message comme non-lu en remettant le marqueur **N**),
- **l** pour n'afficher que certains messages de la boîte aux lettres, tapez alors :
  - **~f toto** pour afficher les messages contenant *toto* dans le champ *From*,
  - **~C titi** pour afficher les messages contenant *titi* dans le champ *To* ou dans le champ *Cc* (taper **~t titi** pour se limiter au champ *To* et **~c titi** pour se limiter au champ *Cc*),
  - **~s urgent** pour afficher les messages contenant *urgent* dans le sujet,
  - **~b bruit** pour afficher les messages contenant le mot *bruit* dans le texte du message,
  - **~F** pour afficher les messages marqués comme important,
  - **~A** pour afficher tous les messages.
- **q** pour quitter Mutt.

Note

Pour revenir à la liste des messages quand vous êtes en train d'afficher un message, tapez **e**.

A tout moment, vous pouvez taper **?** pour accéder à la liste des touches disponibles avec leur fonction.

46.4 C'est parti !

Maintenant que tout est en place, vous pouvez :

1. lancer le rapatriement de vos messages :

```
% fetchmail
```

2. vérifier qu'ils sont correctement triés en consultant le fichier `~/.procmail.log` ;
3. les lire avec *mutt* :

```
% mutt -y
```

Quand vous envoyez un message avec mutt, il est transmis à Postfix qui essaye alors de joindre le serveur du destinataire, ou votre serveur relai si vous en avez défini un. Si ce serveur n'est pas joignable ou si vous n'êtes pas connecté à Internet, le message reste alors stocké dans la queue de messages de *Postfix*.

Pour voir les messages en attente dans la queue de *Postfix* :

```
% mailq
```

Pour effacer un mail stocké dans la queue :

```
# postsuper -d ID_du_message
```

où *ID_du_message* est l'ID du message tel qu'il apparaît avec la commande **mailq**.

Postfix essaye de vider sa queue à intervalles réguliers. Pour le forcer à vider sa queue immédiatement :

```
% postqueue -f
```

Pour vous rendre la vie plus facile, j'ai créé des raccourcis clavier vers ces commandes dans Mutt (cf `.muttrc`) :

- **F2** pour exécuter **mailq**,
- **F3** pour exécuter **postqueue -f**,
- **F4** pour exécuter **fetchmail**.

Chapitre 47

Le courrier en console (suite)

47.1 Archiver son courrier

Principe

Au bout d'un certain temps, si vous recevez beaucoup de courrier, vos boîtes aux lettres peuvent commencer à devenir trop grosses, ce qui a deux inconvénients : ouvrir une boîte aux lettres (ou la synchroniser, comme expliqué ci-dessous) prend beaucoup de temps, et le courrier commence à prendre beaucoup d'espace disque.

Pour remédier à ce problème, je vous propose d'utiliser le programme archivemail pour :

- supprimer les messages trop vieux de certaines boîtes aux lettres (par exemple celles contenant des mailing-lists archivées sur le Web) ;
- compresser les messages trop vieux de certaines boîtes aux lettres.

Mise en œuvre

Commencez par installer le programme archivemail :

```
# aptitude install archivemail
```

La sytaxe d'archivemail est très simple :

- pour supprimer les messages vieux de plus de 90 jours des boîtes aux lettres `mailbox1` et `mailbox2` :

```
% archivemail -d 90 --delete ~/mail/mailbox1 ~/mail/ ↩
    mailbox2
```

- pour compresser les messages vieux de plus de 120 jours des boîtes aux lettres `mailbox3` et `mailbox4` et les stocker au format *mbox* dans le répertoire `~/mail-archive` préalablement crée :

```
% archivemail -d 120 -o ~/mail-archive ~/mail/mailbox3 ↩
    ~/mail/mailbox4
```

Si les commandes précédentes ne marchent pas et génèrent un message d'erreur bizarre, il faut relancer la commande en ajoutant l'option **-v** (verbose) et regarder quel message particulier est responsable de l'erreur ; ensuite, éditer ce message fautif pour

essayer de comprendre pourquoi il empêche archivemail de fonctionner, et supprimer le à la main s'il n'est pas important.

Si vous décidez de mettre en place une telle solution, vous aurez probablement envie que cela se fasse automatiquement, par exemple chaque lundi à 12h42. Pour cela, vous pouvez utiliser la commande *crontab*, comme expliqué au Chapitre 51 et résumée ici. Editez votre *cron* avec la commande **crontab -e** et rajoutez lres deux lignes suivantes :

```
42 12 * * 1 archivemail --quiet -d 90 --delete ~/mail/ ↩
    mailbox1 ~/mail/mailbox2
43 12 * * 1 archivemail --quiet -d 120 -o ~/mail-archive ~/ ↩
    mail/mailbox3 ~/mail/mailbox4
```

47.2 Synchroniser ses messages entre un serveur et un portable

Le principe

C'est la configuration que j'utilise personnellement ; je vais donc parler à la première personne ! Je reçois mon courrier sur un serveur sous Debian sur lequel j'ai un accès par SSH. Sur ce serveur, je trie mes messages avec Procmail et SpamAssassin et je les lis avec Mutt, exactement comme expliqué au Chapitre 46. Sur ce serveur, je fais également l'archivage de mes messages comme expliqué dans la section précédente. Avec cette solution, je peux lire mes messages depuis n'importe quel ordinateur connecté à Internet et sur lequel je peux installer un client SSH.

J'ai aussi un ordinateur portable sous Debian, et je veux pouvoir garder mes messages sur mon portable même quand il n'est pas connecté. En plus de ça, je veux :

- pouvoir synchroniser facilement toutes mes boîtes aux lettres entre le serveur et mon portable de manière sécurisée (c'est à dire sans que le mot de passe soit envoyé en clair sur le réseau ni stocké en clair dans un fichier texte sur mon portable) ;
- que les messages que je lis sur mon portable et ceux que je lis sur le serveur soient marqués comme lus sur mon portable *et* sur le serveur après synchronisation ;
- que les messages que je marque pour être effacés sur mon portable et ceux que je marque pour être effacés sur le serveur soient effacés sur mon portable *et* sur le serveur après synchronisation ;
- que les messages auxquels je réponds sur mon portable et ceux auxquels je réponds sur le serveur soient marqués comme répondus sur mon portable *et* sur le serveur après synchronisation.

C'est le programme isync qui va s'occuper de la synchronisation. La mise en place de ma solution avec isync requiert trois choses :

- un accès SSH au serveur ;
- un serveur IMAP installé sur le serveur (mais pas forcémment joignable de l'extérieur) ;
- des boîtes aux lettres au format *Maildir* sur le serveur.

Note
Si vous avez suivi les instructions du Chapitre 46, vos boîtes aux lettres sont au format *Maildir*.

Faire une sauvegarde

Dans ce genre de bidouilles, une mauvaise manipulation est vite arrivée : je vous conseille donc de faire une sauvegarde de vos messages avant de commencer.

Sur le serveur

Il faut qu'un serveur IMAP soit installé sur le serveur et repérer l'exécutable du serveur IMAP. Sur une Debian qui utilise le serveur IMAP fourni dans le paquet *uw-imapd*, l'exécutable du serveur est `/usr/sbin/imapd`.

Ensuite, il faut modifier la configuration de mutt sur le serveur. Sur le serveur, éditez le fichier de configuration `~/.muttrc` et, si vous utilisez mon fichier de configuration, décommentez la ligne suivante, située dans la première partie :

```
set maildir_trash
```

Sur le portable

Je suppose que le portable n'a pour l'instant aucune configuration de faite au niveau mail.

Mutt est normalement déjà installé. Récupérez mon fichier de configuration type :

```
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
   muttrc
% mv muttrc ~/.muttrc
```

Editez le fichier `~/.muttrc`. Commencez par adapter la première partie intitulée *Configuration Générale* et décommentez la ligne suivante :

```
set maildir_trash
```

Ensuite, dans la partie intitulée *Binding des touches*, dans le paragraphe sur les macros, vous pouvez éventuellement commenter les lignes qui associent la touche **F4** avec fetchmail et décommenter les lignes qui associent **F4** avec isync.

Créez le sous-répertoire `mail/` dans votre répertoire personnel pour accueillir vos boîtes aux lettres :

```
% mkdir ~/mail/
```

Installez isync :

```
# aptitude install isync
```

Récupérez mon fichier de configuration type pour isync :

```
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
   isyncrc
% mv isyncrc ~/.isyncrc
```

Editez le fichier `~/.isyncrc` et adaptez les lignes commençant par :

– *Host* : mettez le nom DNS de votre serveur ;

```
Host serveur.exemple.org
```

– *Tunnel* : mettez votre login, le nom DNS de votre serveur et l'exécutable du serveur IMAP sur votre serveur.

```
Tunnel "ssh -q toto@serveur.exemple.org /usr/sbin/imapd"
```

Synchroniser

Tout est maintenant prêt pour la première synchronisation ! Sur votre portable, vous allez lancer **isync** avec l'option **-a** pour synchroniser toutes les boîtes aux lettres, l'option **-L** pour créer automatiquement les boîtes aux lettres en local et l'option **-V** (verbose) pour faciliter le diagnostic des éventuels problèmes. Si en plus vous avez mis en place *ssh-agent* comme expliqué dans le Chapitre 40, exécutez **ssh-add** préalablement pour ne même pas avoir à taper votre mot de passe !

```
% isync -a -L -V
```

Si vous avez beaucoup de messages sur le serveur, cela risque de prendre un peu de temps. Pour les prochaines synchronisation, vous pourrez exécuter :
– **isync -a** pour synchroniser toutes les boîtes aux lettres ;
– **isync mailbox1 mailbox2** pour ne synchroniser que certaines boîtes aux lettres.
et ce sera plus rapide puisqu'il n'aura que les nouveaux messages à télécharger.

Note
Un grand merci à Oswald Buddenhagen, nouveau mainteneur d'isync, pour m'avoir aidé à faire marcher cette solution.

47.3 Chiffrer et signer ses mails avec GnuPG

A l'adresse http://www.vilya.org/gpg/gpg-intro.html, vous trouverez une documentation en français très bien faite sur GnuPG, son utilisation et son intégration à *mutt*.

47.4 Adapter le champ From: en fonction du destinataire

Il est souvent nécessaire de devoir adapter l'adresse électronique que l'on utilise en fonction de l'adresse à laquelle on écrit. Par exemple, je peux avoir envie d'utiliser mon adresse @via.ecp.fr à chaque fois que j'écris à quelqu'un qui a une adresse @via.ecp.fr.

```
send-hook '~t ^.*@via\.ecp\.fr$' 'set from="robert.dupont@via ↩
    .ecp.fr"'
```

De même, je peux vouloir utiliser mon adresse professionnelle à chaque fois que j'écris à un collègue. Il suffit alors d'ajouter des entrées de ce type pour chaque collègue (remplacer toto@free.fr par l'adresse de vos collègues) :

```
send-hook '~t ^toto@free\.fr$' 'set from="robert.dupont@mon. ↩
    employeur.fr"'
```

Ces deux exemples sont commentés à la fin de la section 3 du fichier de configuration de Mutt que nous vous avons fourni.

Chapitre 48

L'IRC en console

48.1 Installer un client IRC

Je vous propose d'installer le client IRC irssi :

```
# aptitude install irssi-text
```

48.2 Utiliser et configurer irssi

Lancer irssi

La configuration d'*irssi* peut se faire à l'intérieur du programme par des commandes ou en éditant son fichier de configuration `~/.irssi/config` (qui sera crée au premier lancement). Lancez donc le programme :

```
% irssi
```

Quand vous lancez *irssi*, vous vous retrouvez dans une fenêtre aux bordures bleues avec un prompt **[(status)]** en bas de la fenêtre, comme sur la Figure 48.1.

```
 Irssi v0.8.12 - http://www.irssi.org
23:21 -!- Irssi: Looks like this is the first time you've run irssi.
23:21 -!- Irssi: This is just a reminder that you really should go read
23:21 -!- Irssi: startup-HOWTO if you haven't already. You can find it
23:21 -!- Irssi: and more irssi beginner info at http://www.irssi.org
23:21 -!- Irssi:
23:21 -!- Irssi: For the truly impatient people who don't like any automatic
23:21 -!- Irssi: window creation or closing, just type: /MANUAL-WINDOWS
23:21 -!- Irssi:
23:21 -!- Irssi: For Debian specific help type "/connect irc.debian.org"
23:21 -!- Irssi: and "/join #debian" (without the quotes) and ask your
23:21 -!- Irssi: question.

 [23:21] [] [1]
[(status)]
```

FIGURE 48.1 – Irssi à son lancement

Vous êtes en fait dans la première fenêtre appelée *status* dans laquelle se font les réglages.

La configuration se fait par des commandes commençant par **/**. Un des gros avantages d'*irssi* est sa capacité à être connecté à plusieurs serveurs IRC en même temps. Pour cela, il faut définir des réseaux, qui correspondent à un serveur IRC seul ou à plusieurs serveurs IRC reliés entre-eux, grâce à la commande **/network add**. Il faut ensuite donner l'adresse d'au moins un serveur IRC par réseau avec la commande **/server add**.

Par exemple, je veux me connecter à deux réseaux :
- *OpenProjects* (alias OPN maintenant appelé FreeNode), qui est déjà configuré par défaut ;
- *ExempleNet* qui est un réseau pour l'exemple...

Pour voir la liste des réseaux déjà définis par défaut :

```
[(status)] /network
```

et il vous affiche la liste suivante dans la fenêtre :

```
23:25 Networks:
23:25 IRCnet: querychans: 5, max_kicks: 4, max_msgs: 5,  ↩
    max_whois: 4
23:25 EFNet: max_kicks: 4, max_msgs: 3, max_whois: 1
23:25 Undernet: max_kicks: 1, max_msgs: 3, max_whois: 30
23:25 DALnet: max_kicks: 4, max_msgs: 3, max_whois: 30
23:25 QuakeNet: max_kicks: 1, max_msgs: 3, max_whois: 30
```

Vous voyez donc qu'*IRCnet* est déjà défini. Pour rajouter le réseau *OFTC* (Open and Free Technology Community), il faut lui choisir un alias (nous choisirons *OFTC*) et définir au moins un de ses serveurs, par exemple *irc.oftc.net* :

```
[(status)] /network add OFTC
[(status)] /server add -auto -network OFTC irc.oftc.net
```

Se connecter aux réseaux IRC

Maintenant que notre réseau est défini, il est très facile de s'y connecter en le désignant par son alias :

```
[(status)] /connect OFTC
```

Vous pouvez vous connecter en même temps à d'autres réseaux en utilisant à nouveau la commande **/connect** avec un autre alias.

Rejoindre des canaux et jouer avec les fenêtres

Pour rejoindre le canal (ou salon) *#debian* sur *OFTC* :

```
[(status)] /join -oftc #debian
```

La fenêtre se redessine et vous entrez dans le canal *#debian* : Figure 48.2. Le sujet du canal s'affiche dans la barre bleue du haut et la liste des personnes présentes apparaît. Vous pouvez alors discuter normalement en bénéficiant de la complétion automatique sur les pseudos.

```
 4.0r4 released /msg dpkg etch | /msg dpkg etch->lenny | PUBLIC KEY NOT AVAILABLE?
                      extensions?
23:29 < Baltazaar> I have posted it to flood.
23:29 < Baltazaar> it says cannot sent to channel
23:29 -!- Zerocxis_ [~sixcorez@h228.21.134.98.ip.windstream.net] has quit [Quit:
          Leaving]
23:29 < _Nameeater> you will have to pastebin it :)
23:30 -!- ajonat [~ajonat@190.48.119.128] has quit [Quit: ajonat]
23:30 < Baltazaar> ok, how do I do that? I use Gaim...
23:30 < interbird> can you switch to #apache-fix ?
23:30 -!- ajonat [~ajonat@190.48.119.128] has joined #debian
23:30 -!- ajonat [~ajonat@190.48.119.128] has quit [Read error: Connection reset
          by peer]
23:30 < petethepirate> interbird sure.
23:30 < _Nameeater> Baltazaar: http://paste.debian.net/
23:31 -!- petethepirate [~qwerty@ip70-177-102-187.ok.ok.cox.net] has left #debian
          []
23:32 < Baltazaar> by the way, sorry for being such a newbie, and wasting so much
                   time.... it is now under
23:32 < Baltazaar>  Posting 18049 from Baltazaar posted at 2008-09-27 05:31:37
                   expires:  2008-09-30 05:31:3
23:32 < _Nameeater> ok, I will take a look
23:33 < Baltazaar> 29 lines of pure and evil errorcode.....
 [23:34] [Elessar(+Ri)] [7:#debian(+clnt 300)] [Act: 2,3,4,5,6]
[#debian] []
```

FIGURE 48.2 – Sur #debian avec irssi

En fait, *irssi* a ouvert une nouvelle fenêtre. Pour naviguer d'une fenêtre à l'autre, utilisez les touches suivantes :

– Ctrl-N : va à la fenêtre suivante ;
– Ctrl-P : va à la fenêtre précédente ;
– Alt-2 : va à la fenêtre n°2.

Avec une de ces combinaisons de touches, revenez à la fenêtre *status*, qui est la seule fenêtre dans laquelle le résultat des commandes apparaît. Pour rejoindre le channel *#test* du réseau IRCnet, utilisez la commande suivante :

```
[(status)] /join -ircnet #test
```

Commandes pour un canal

– **/me agit** : envoie un message d'action (dans lequel */me* apparaîtra remplacé par votre pseudonyme) sur le channel,
– **/nick nouveau_nick** : change de pseudonyme,
– **/query nick** : entamme une conversation privée dans une nouvelle fenêtre,
– **/query** : met fin à la conversation privée et ferme la fenêtre,
– **/leave** : quitte le canal et ferme la fenêtre.

Commandes pour la fenêtre de statut

Les commandes suivantes peuvent être tapées dans n'importe quelle fenêtre mais leur résultat sera écrit dans la fenêtre *status*. Certaines commandes concernent un certain réseau ; pour passer d'un réseau à l'autre, on utilise la combinaison de touches Ctrl-X pour cycler sur les *ircnets* auquels vous êtes connecté. Le nom du réseau sélectionné apparaît dans la barre bleue du bas de la fenêtre.

– **/list** : affiche la liste des canaux du réseau sélectionné,
– **/who #debian** : affiche la liste des gens présents sur #debian,

- **/hilight mot** : met en couleur le pseudo de la personne qui prononce le mot (c'est utile pour repérer quand on parle de soi),
- **/dcc list** : liste les fichiers disponibles par dcc,
- **/dcc get nick** : pour récupérer un fichier envoyé par dcc,
- **/dcc send nick nom_du_fichier** : pour envoyer un fichier par dcc,
- **/help** pour avoir la liste des commandes,
- **/help nom_de_la_commande** pour avoir de l'aide sur une commande particulière,
- **/quit** pour quitter *irssi*.

L'indicateur d'activité

Dans la barre bleue du bas, il y a un indicateur d'activité des différentes fenêtres : Figure 48.3.

[23:50] [Elessar(+Ri)] [7:#debian(+clnt 293)] [Act: 2,5,6]

FIGURE 48.3 – Barre d'activité d'irssi

Si on prend l'exemple de la barre bleue ci-dessus, on peut voir, de gauche à droite :
- l'heure,
- mon pseudo (*Elessar*),
- le nom du canal (*#debian*) et le numéro de la fenêtre (*7*),
- l'indicateur d'activité : les numéros en blanc correspondent aux numéros des autres fenêtres dans lesquelles il y a de l'activité (*4* et *6*), et les numéros en violet correspondent aux numéros des autres fenêtres où quelqu'un a prononcé mon nick ou où quelqu'un me parle en privé.

Bip !

Si vous souhaitez que quelqu'un puisse facilement vous joindre sur l'IRC, vous avez peut-être envie que votre terminal bippe quand on prononce votre pseudo, quand on vous parle en privé ou quand on vous envoie un message par DCC. Pour cela, tapez les commandes suivantes dans la fenêtre *status* :

```
[(status)] /set beep_when_window_active on
[(status)] /set beep_when_away on
[(status)] /set bell_beeps on
[(status)] /set beep_msg_level HILIGHT MSGS DCC DCCMSGS
[(status)] /hilight ton_nick
```

Astuce

Je vous propose d'utiliser les touches **F1**, **F2**,... jusqu'à **F12** pour zapper facilement des fenêtres 1 à 12. Pour cela, sauvegardez la configuration actuelle :

```
[(status)] /save
```

Puis, en utilisant une autre console, rajoutez à la fin du fichier de configuration `~/.irssi/config` les lignes contenues dans le fichier `irssi-touches-fonction` :

```
% cat /root/config/irssi-touches-fonction >> ~/.irssi/config
```

ou :

```
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
    irssi-touches-fonction
% cat irssi-touches-fonction >> ~/.irssi/config
```

Puis demande à *irssi* de relire son fichier de configuration :

```
[(status)] /reload
```

Chapitre 49

Les news en console

49.1 Installer et configurer slrn

Installer le paquet

Je vous propose d'installer le client news en mode texte slrn :

```
# aptitude install slrn
```

Configurer slrn

Dès l'installation, il vous demande l'adresse de votre serveur de news. Ce paramètre est en fait l'adresse du serveur de news par défaut, mais chaque utilisateur pourra s'il le désire modifier ce paramètre.

Chaque utilisateur a son propre fichier de configuration `.slrnrc` dans son home. Ce fichier est lu à chaque lancement de *slrn*. Je vous propose d'utiliser mon fichier de configuration type et de le mettre dans votre home sous le nom `.slrnrc` :

```
% cp /root/config/slrnrc ~/.slrnrc
```

ou :

```
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
   slrnrc
% mv slrnrc ~/.slrnrc
```

Editez-le pour le personnaliser en vous aidant des commentaires. Les champs que vous devez absolument personnaliser sont les suivants :

- **set username**
- **set hostname**
- **set realname**
- **set organization**
- **server**

Une fois le fichier de configuration au point, créez le répertoire `.news/` et à l'intérieur un fichier `score` vide :

```
% mkdir ~/.news
% touch ~/.news/score
```

Note

Si vous voulez utiliser un serveur de news différent du serveur de news définit par défaut (dont l'adresse est stockée dans le fichier `/etc/news/server`), il faut stocker son adresse dans la variable d'environnement *NNTPSERVER*.

49.2 Utiliser slrn

Au premier lancement de *slrn*, vous devez ajouter une option pour créer le fichier appelé "*newsrc*" dans lequel seront stockées les informations sur les forums :

```
% slrn -f ~/.jnewsrc-serveur --create
```

où `.jnewsrc-serveur` est le nom de fichier que vous avez indiqué comme deuxième paramètre du champ **server** de votre `slrnrc`.

Pour les prochains lancements, il vous suffira de taper simplement :

```
% slrn
```

Une fois que *slrn* est lancé, il vous présente un bel écran presque vide. C'est normalement l'écran dans lequel apparaît la liste des forums auquels vous êtes abonné. Tapez **L** puis par exemple **fr.comp*** et **Entrée** pour obtenir la liste des forums en français qui parlent d'informatique. Il y a un **U** (comme *Unsubscribe*) à leur gauche, ce qui signifie que vous n'êtes pas abonné. Pour vous abonnez aux forums qui vous intéressent, placez le curseur à côté du nom du forum et tapez **s** (comme *Subscribe*). Le **U** disparaît alors. Pour se désabonner, il suffit de taper **u**. Le nombre à gauche du nom des forums indique le nombre de messages non marqués comme *lu* dans ce forum.

```
slrn 0.9.7.4 ** Pressez '?' pour l'aide, 'q' pour quitter. ** Serveur : news.ecp
  U  340  fr.comp.applications.emacs
  U  195  fr.comp.applications.groupware
     258  fr.comp.applications.libres
     129  fr.comp.applications.x11
     380  fr.comp.developpement
  U 2611  fr.comp.divers
  U 1539  fr.comp.emulateurs
  U  388  fr.comp.ia
  U  267  fr.comp.infosystemes
  U 1933  fr.comp.infosystemes.www.auteurs
  U  101  fr.comp.infosystemes.www.divers
  U 2306  fr.comp.infosystemes.www.navigateur
  U 1921  fr.comp.infosystemes.www.pages-pers
  U  575  fr.comp.infosystemes.www.serveurs
  U  111  fr.comp.lang.ada
  U  704  fr.comp.lang.basic
    2341  fr.comp.lang.c
  U 3438  fr.comp.lang.c++
  U  150  fr.comp.lang.general
  U 1849  fr.comp.lang.java
  U   27  fr.comp.lang.lisp
  U  214  fr.comp.lang.pascal
  U  767  fr.comp.lang.perl
  U  282  fr.comp.lang.tcl
    2080  fr.comp.mail
->U 2238  fr.comp.musique
  U   78  fr.comp.objet
  U 1287  fr.comp.os.bsd
-*-News Groups: news.ecp.fr                              -- 26/85 (Haut)
Esp:Sél, p:Poster c:Marquer-lu l:Lister ^R:Redessiner (u)s:(Dés)abonner
```

FIGURE 49.1 – Fenêtre des forums de slrn

Maintenant que vous avez choisi les forums auquels vous voulez être abonné, mettez-vous en face de l'un d'entre eux et appuyez sur **Espace**. Vous avez alors accès à la liste des threads, précédés du nombre de messages qu'ils contiennent. Pour ouvrir un thread et lire le premier message qu'il contient, appuyez de nouveau sur **Espace**. Pour faire défiler le message vers le bas, appuyez sur **Espace** ; pour le faire défiler vers le haut, appuyez sur **b**. Pour passer à la lecture du message suivant, appuyez sur **Espace** une fois que vous êtes arrivé à la fin du message, ou sélectionnez avec la flèche le message suivant et appuyez sur **Espace**.

```
slrn 0.9.7.4 ** Pressez '?' pour l'aide, 'q' pour quitter. ** Serveur : news.ecp
 1  -         43:[Gildas Le Go]    [Annonce] Hamster-Fr 2.0.1.8
 2  -         13:[Gilles      ]  6 browser d'image
 3  D         40:[Matthieu Moy]    MP3 non libre :-( => passez au format OGG !
->  D         81:[Kyle Gardner]     |->
 [254/256 non-lus] Forum: fr.comp.applications.libres        -- 4/61 (Haut)
From: Kyle Gardner <kylegl@techemail.com>
X-Mailer: Mozilla 4.7 [en] (WinNT; I)
Newsgroups: fr.comp.applications.libres
Subject: Re: MP3 non libre :-( => passez au format OGG !
Organization: LORIA & INRIA-Lorraine - Nancy - FRANCE
Date: 04 Sep 2002 19:11:18 GMT

Matthieu Moy wrote:
> En fait, le codage comme le décodage sont soumis à des brevets détenus
> par Thomson, [...]

Brevets dont la validite reste a prouver en Europe.
En effet, ce n'est pas parce que le brevet a ete accorde par
l'INPI (en France) ou l'EPO (European Patent Office - pour l'europe)
qu'il est valide. La validite du brevet est determinee devant une
cours nationale. Et pour l'instant, toutes les decisions de justice concernant
les brevets logiciels ont invalide les brevets accordes par l'EPO.
Cela est particulierement vrai en allemagne :

- http://swpat.ffii.org/papers/bpatg17-suche00/
  2000 - court allemande contre IBM -
  Precise qu'un programme informatique "en tant que tel"
  n'est rien d'autre qu'un programme informatique, et
 2619 : Re: MP3 non libre :-( => passez au format OGG !      -- 1/88 (Haut)
Esp:PagSuiv  B:PagPré u:Marquer-non-lu  f:Poursuivre n:suivant p:Préc
```

FIGURE 49.2 – Fenêtre des articles de slrn

Pour revenir à la liste des messages du forum, appuyez sur **h**. Pour marquer un message ou un thread comme *lu* sans même le lire, appuyez sur **d** en face de ce message ou de ce thread.

Pour revenir à la liste des forums, appuyez sur **q**. Pour rapatrier depuis le serveur la liste des nouveaux posts, appuyez sur **G**.

Entraînez-vous à poster avec *slrn* dans un forum prévu à cet effet : *fr.test*. Pour cela, commencez par souscrire à ce forum comme décrit ci-desssus. Ensuite, sélectionnez ce forum et appuyez sur **P** : il vous demande en bas de l'écran si vous êtes bien sûr de vouloir publier, puis il vous demande de confirmer le nom du forum, et enfin de définir un sujet. Vous arrivez alors dans votre éditeur de texte favori (si vous n'aimez pas *vim*, vous pouvez en changer dans le fichier de configuration).

Si vous voulez faire un *crosspost*, rajoutez un deuxième nom de forum dans le champ *Newsgroups:* (par exemple *edu.fr.test*). Dans notre exemple, cela donne la ligne :

```
Newsgroups: fr.test,edu.fr.test
```

Pour mettre un Followup-to vers *fr.test* par exemple, il faut mettre le nom du forum dans le champ *Followup-To:* :

```
Followup-To: fr.test
```

Une fois le message écrit, il vous suffit de l'enregistrer et de quitter. Vous revenez alors dans *slrn* qui vous demande confirmation avant de poster.

Pour répondre à un message, placez-vous sur ce message et appuyez sur **r** ; pour répondre par mail perso, appuyez sur **f** (dans la configuration par défaut, c'est l'inverse, mais je préfère comme ça !). Vous vous retrouvez alors une nouvelle fois dans votre éditeur de texte favori...

Pour connaître la liste complète des commandes, il suffit de taper **?** dans *slrn*. Vous y apprendrez par exemple que la combinaison de touches **Echap** puis Ctrl-c permet d'annuler un message que vous avez posté.

Chapitre 50

La messagerie instantanée avec Finch

Dans le Chapitre 32, nous vous avons initié à Pidgin, un logiciel graphique de messagerie instanéee. Il existe également des logiciels de messagerie instantanée utilisables en console. Parmi eux, Finch se taille une place de choix : il s'agit tout simplement d'une version de Pidgin adaptée à la console. Tout comme Pidgin, Finch permet donc d'utiliser un très grand nombre de protocoles de messagerie.

50.1 Installation et prise en main

Installer Finch

Installez simplement le paquet éponyme :

```
# aptitude install finch
```

Principes d'utilisation

Finch peut être lancé avec la commande **finch**, et s'utilise de la même façon que Pidgin : vous y retrouverez donc les mêmes menus, et ces deux logiciels partagent leur configuration.

L'interface de Finch utilise des *fenêtres semi-graphiques*, dessinées à l'aide de la bibliothèque *ncurses* dans votre console, comme sur la Figure 50.1.

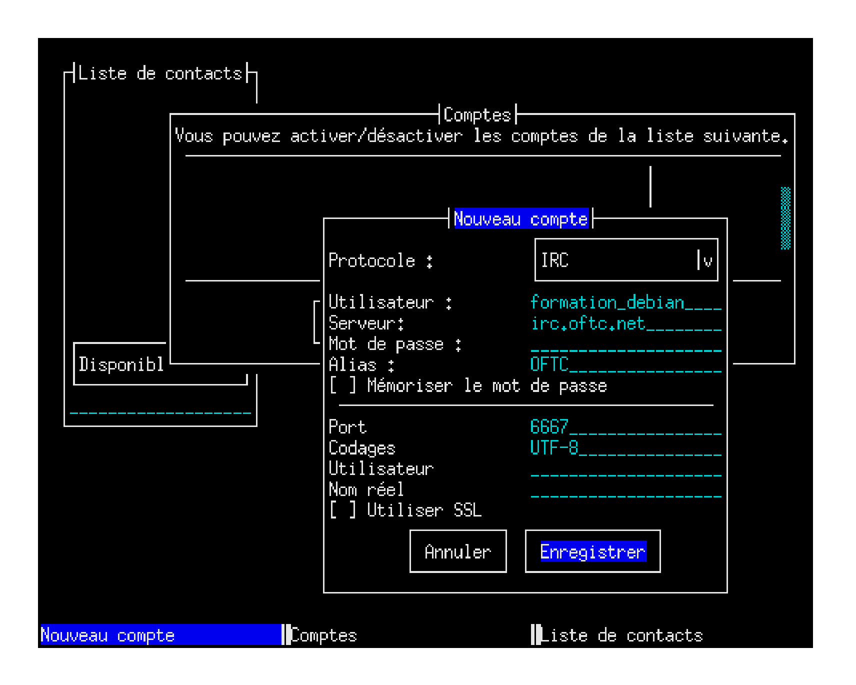


FIGURE 50.1 – Premier lancement de Finch

Note

La Figure 50.1 montre les fenêtres affichées lors du *premier* lancement de Finch. Une fois que vous avez configuré vos comptes de messagerie instantannée, que ce soit avec Finch ou avec Pidgin, seule la fenêtre de liste des contats vous est proposée.

L'interface de Finch se contrôle au clavier, à l'aide des raccourcis suivants (vous pouvez retrouver la liste complète dans le manuel **man finch**) :

Alt-n et Alt-p (*next, previous*) changer de fenêtre dans un sens ou dans l'autre

Alt-m (*move*) puis les flèches déplacer la fenêtre sélectionnée (**Échap** permet de quitter ce mode)

Alt-r (*resize*) de la même façon, redimentionner la fenêtre

Alt-c (*close*) fermer la fenêtre sélectionnée

Alt-q quitter Finch

F10 affiche le menu de la fenêtre sélectionnée

F11 affiche le menu contextuel de l'entrée sélectionnée

Tab dans un formulaire, change de champ

50.2 Configuration

Si vous n'avez pas encore défini de compte, vous pouvez remplir la fenêtre de configuration d'un nouveau compte, en choisissant le protocole de messagerie, le serveur, ainsi que des paramètres spécifiques au protocole choisi : Figure 50.1.

Après avoir validé, vous vous retrouvez dans la fenêtre des comptes (Figure 50.2), et vous pouvez activer le compte que vous venez de régler avec la touche **Espace**.



FIGURE 50.2 – La liste des comptes

50.3 Utilisation

L'utilisation de Finch est centrée sur la *liste des contacts* : fermez la fenêtre des comptes avec Alt-c (vous pourrez la rouvrir depuis le menu, avec **F10**), pour sélectionner cette liste.

Vous pouvez maintenant ajouter des contacts depuis le menu (**F10**) *Options* > *Ajouter* ou depuis le menu contextuel (**F11**). Pour chaque contact, vous devez définir le compte et les informations permettant d'ouvrir une discussion, comme sur la Figure 50.3.

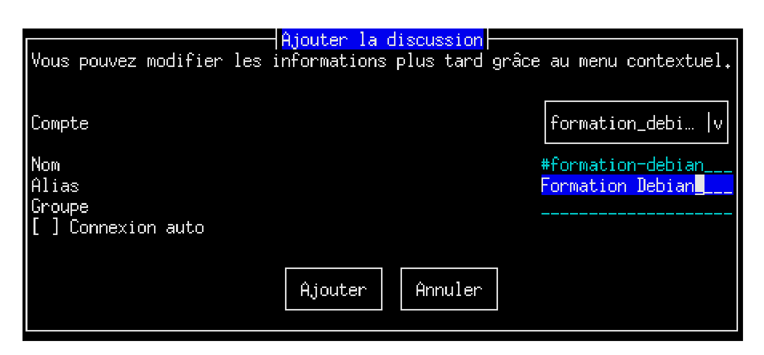


FIGURE 50.3 – Ajout d'un contact : un canal de discussion

Après avoir défini vos contacts, vous pouvez ouvrir des discussions en les sélectionnant et en appuyant sur **Entrée**. Pour terminer une discussion, fermez simplement la fenêtre correspondante avec Alt-c. Pour quitter Finch, utilisez la combinaison Alt-q.

Note

Finch semble affecté d'un bogue assez gênant : lorsqu'on l'utilise depuis un émulateur de terminal comme celui de GNOME ou xterm, il est impossible de saisir des caractères accentués. Malheureusement, les logiciels concurrents ont des défauts pire encore… Il reste possible de saisir des caractères accentués en utilisant Finch dans un rxvt-unicode (paquet *rxvt-unicode*), par exemple.

Chapitre 51

Outils d’administration système

51.1 La programmation de tâches

Exécuter une commande périodiquement

La cron est un programme (installé par défaut) qui est chargé de lancer d’autres programmes de manière périodique et automatique. Chaque utilisateur peut définir avec sa *crontab* les programmes qu’il veut lancer périodiquement. Il lui suffit d’éditer sa *cron* et de définir la commande et sa périodicité d’exécution.

Par exemple, je veux que mon ordinateur me réveille tous les matins à 7h12 en jouant un mp3. J’édite ma crontab :

```
% crontab -e
```

Je me retrouve alors dans *vim* avec un fichier vide. J’ajoute la ligne :

```
12 7 * * *  music123 ~/music/fichier.mp3
```

J’enregistre et je quitte ; les changements sont alors automatiquement pris en compte par le système.

Explications :

1. Les 5 premiers ensembles de caractères séparés par des espaces (ici, *12 7 * * **) définissent la fréquence. Dans l’ordre, on trouve :
   – les minutes,
   – les heures,
   – le jour du mois,
   – le mois,
   – le jour de la semaine (sachant que Lundi vaut 1, Mardi vaut 2, etc.).
   Pour exécuter une commande chaque jour à 7h12, on fixe donc le champ minute à 12, le champ heure à 7, puis on met des étoiles dans les autres champs pour dire qu’il faut que ce soit exécuté tous les jours dans le mois, tous les mois et tous les jours de la semaine.
2. Enfin, on trouve la commande à exécuter : (ici, *music123 ~/music/fichier.mp3*).

Quand la cron lance un programme, elle envoie par mail à l’utilisateur le texte que ce programme écrirait sur la console s’il était lancé à la main ; sauf si le programme n’a rien écrit. Pour éviter de recevoir un mail tous les matins qui me dit qu’il a bien

joué mon fichier mp3, je modifie l'entrée dans la cron en renvoyant la sortie texte du programme vers `/dev/null` :

```
12 7 * * *  music123 ~/music/fichier.mp3 1>/dev/null 2>&1
```

Autre exemple : j'ai une connexion permanente à Internet et je veux aussi que fetchmail aille chercher mes mails tous les quarts d'heure. J'édite ma cron et je rajoute la ligne suivante :

```
*/15 * * * * fetchmail --silent
```

Explications :

1. Les 5 premiers ensembles de caractères **/15* * * * * définissent la fréquence « toutes les quinze minutes ».
2. J'exécute fetchmail avec l'option *--silent* qui n'écrit dans sa sortie que les messages d'erreur ; comme ça, je ne recevrais un mail que quand le rappatriement des mails se passe mal.

Exécuter une commande à une date donnée

Par exemple, pour exécuter les commandes **date** puis **df -h** le 28 septembre 2008 à midi 42, tapez :

```
% at 12:42 28.09.2008
```

Vous voyez alors apparaître le prompt du programme *at* pour taper les commandes :

```
at> date
at> df -h
```

Une fois que vous avez fini de taper les commandes, utilisez la combinaison de touches Ctrl-d. Le résultat de la série de commandes, appelée *job*, vous sera envoyé par mail juste après leur exécution.

Pour voir la liste des jobs en attente, utilisez la commande **atq**. Pour annuler un job, tapez **atrm numéro_du_job**.

51.2 Donner des droits étendus à certains utilisateurs

Vous avez peut-être déjà remarqué, certaines commandes pourtant courantes ne sont accessibles qu'à l'utilisateur *root*, par exemple les commandes pour éteindre l'ordinateur, pour le rebooter ou encore le mettre en veille. Si cela se comprend pour un serveur, c'est souvent gênant pour un ordinateur familial. La mauvaise solution serait de donner le mot de passe *root* à tout le monde ; la bonne solution est d'utiliser le programme sudo qui permet au *root* de spécifier que certains utilisateurs ont le droit d'exécuter certaines commandes avec les droits de *root*.

Attention

La configuration du programme sudo doit se faire avec attention, pour ne pas compromettre la sécurité du système. Par exemple, si vous donnez à un utilisateur le droit d'exécuter le programme `/bin/sh` en tant que root, alors cela revient à lui donner les droits root tout entiers, car il pourra obtenir un shell avec les droits de *root*. Plus subtil : si vous donnez à un utilisateur le droit d'exécuter `vim un_certain_fichier` en tant que root, cela revient également à lui donner les droits de root tout entiers, car il peut alors ouvrir d'autres fichiers en tant que root en tapant en mode commande **:split autre_fichier** ou encore exécuter des commandes shell en tant que root en tapant en mode commande **:! commande_shell** ; il vaut donc mieux dans ce cas changer les permissions sur le fichier en question.

Maintenant que vous êtes prévenu, vous pouvez installer le paquet :

```
# aptitude install sudo
```

La configuration se fait dans le fichier `/etc/sudoers`. Mais attention, il ne faut pas éditer ce fichier directement avec un éditeur de texte, mais utiliser le programme **visudo**, qui vérifie que vous n'avez pas fait d'erreur lorsque vous enregistrez les nouveaux réglages :

```
# visudo
```

Vous vous retrouvez alors dans un Vim normal entrain d'éditer le fichier `/etc/sudoers`. Pour donner à l'utilisateur *toto* le droit d'exécuter les commandes **halt**, **reboot** et **apm**, rajoutez la ligne suivante :

```
toto  ALL = NOPASSWD: /sbin/halt, /sbin/reboot, /usr/bin/apm
```

Enregistrez et quittez, comme avec un Vim normal. L'utilisateur *toto* peut alors éteindre le système en tapant :

```
% sudo halt
```

Si vous voulez obliger les utilisateurs à re-taper leur mot de passe quand ils utilisent le programme **sudo** (pour être sûr que quelqu'un n'est pas entrain de profiter d'une console laissée ouverte) enlevez de la ligne le mot *NOPASSWD:* .

Pour en savoir plus sur la syntaxe du fichier `sudoers`, consultez `/usr/share/doc/sudo/examples/sudoers`.

51.3 Les outils de compression

Installer les paquets

Le paquet *gzip* qui supporte le *.gz* est installé par défaut, mais ce n'est pas forcément le cas des paquets qui concernent les `.bz2`, `.lzma`, `.xz` et `.zip` :

```
# aptitude install bzip2 zip unzip xz-utils
```

Décompresser un fichier ou un arborescence

La commande à utiliser dépend de l'extension du fichier :

```
% tar -xf archive.tar
% tar -xf archive.tar.gz
% tar -xf archive.tar.bz2
% tar  -xf archive.tar.lzma
% tar  -xf archive.tar.xz
% gunzip archive.gz
% bunzip2 archive.bz2
% unlzma archive.lzma
% unxz archive.xz
% unzip archive.zip
```

Compresser un fichier

Le but de ce paragraphe n'est pas de comparer la performance des différents algorithmes de compression, mais simplement de donner les commandes :

– pour produire un fichier d'extension `.gz` :

```
% gzip fichier1
```

– pour produire un fichier d'extension `.bz2` :

```
% bzip2 fichier1
```

– pour produire un fichier d'extension `.lzma` :

```
% lzma fichier1
```

– pour produire un fichier d'extension `.xz` :

```
% xz fichier1
```

– pour produire un fichier d'extension `.zip` :

```
% zip fichier1.zip fichier1
```

Compresser une arborescence

Par exemple, je veux comprimer en un seul fichier le contenu du répertoire `/etc/` avec en plus le fichier `/var/log/syslog` :

– pour produire un fichier d'extension `.tar.gz` :

```
% tar -acf archive.tar.gz /etc/ /var/log/syslog
```

– pour produire un fichier d'extension `.tar.bz2` :

```
% tar -acf archive.tar.bz2 /etc/ /var/log/syslog
```

– pour produire un fichier d'extension `.tar.lzma` :

```
% tar -acf archive.tar.lzma /etc/ /var/log/syslog
```

– pour produire un fichier d'extension `.tar.xz` :

```
% tar -acf archive.tar.xz /etc/ /var/log/syslog
```

– pour produire un fichier d'extension `.zip` :

```
% zip -r archive.zip /etc/ /var/log/syslog
```

51.4 Les outils réseau

Les outils réseau suivant peuvent être utiles pour résoudre un problème ou surveiller le traffic réseau (la plupart ne sont accessibles qu'en root) :

– **traceroute**, qui se trouve dans le paquet du même nom, et qui permet de dessiner la route que prennent les packets pour aboutir à une machine distance ;
– **tcpdump**, qui se trouve dans le paquet du même nom, et qui permet d'afficher tout le traffic qui passe par l'interface réseau ;
– **iptraf**, un moniteur de traffic et de débit ;
– les outils du paquet **netdiag**, qui permettent de diagnostiquer un problème de connexion.

Chapitre 52

Graver en console

52.1 Installation des paquets

Installez les paquets requis :

```
# aptitude install genisoimage wodim
```

52.2 Les CD de données

Créer l'image ISO du CD

À partir de fichiers

Avant de commencer à graver, il faut créer l'image ISO du CD, c'est à dire le système de fichiers qui sera gravé tel quel sur le disque. Pour cela, on utilise la commande **genisoimage**.

Prenons l'exemple d'une sauvegarde du home de l'utilisateur *toto* :

1. vérifiez qu'il fait moins que la capacité du CD-R ou RW :

```
% du -sh /home/toto/
```

2. créez l'ISO :

```
% genisoimage -R -r -J --hide-rr-moved -V "SauvHome" -o ↩
    backup.iso /home/toto/
```

Explication des options utilisées dans **genisoimage**:

- **-R** : rajoute l'extension Rock Ridge qui permet de conserver les noms longs et les permissions sous Linux,
- **-r** : permet de générer une image qui soit lisible par tout le monde malgré les permissions activées par l'extension Rock Ridge,
- **-J** : active l'extension Joliet pour avoir les noms longs sous Windows,
- **--hide-rr-moved** : renomme le répertoire `rr_moved` (dû aux extensions Rock Ridge) en un répertoire caché `.rr_moved`,
- **-V** : spécifie un Nom de Volume (ici "*SauvHome*"),
- **-o** : spécifie le nom et l'emplacement du fichier ISO (ici `backup.iso` dans le répertoire courant).

Autre exemple : vous voulez créer une ISO qui contienne le répertoire `/home/toto/`, le fichier `/etc/X11/XF86Config-4`, et en plus le contenu du répertoire `/mnt/win/toto/` qui devra apparaître sur le CD dans un répertoire `win-toto` :

```
% genisoimage -R -r -J --hide-rr-moved -V "SauvHome" -o ↩
    backup.iso
-graft-points /home/toto/ /etc/X11/XF86Config-4 win-toto/=/ ↩
    mnt/win/toto/
```

À partir d'un CD

Pour créer l'image ISO d'un CD de données, il suffit de copier directement les données du disque original vers un fichier :

```
% dd if=/dev/cdrom of=backup.iso
```

Tester l'ISO

Pour vérifier que l'ISO du CD marche bien, on va monter le fichier contenant l'ISO en *loopback* dans le répertoire `/mnt/tmp/` (à créer avant s'il n'existe pas) :

```
# mount -t iso9660 -o loop backup.iso /mnt/tmp
```

Si tout va bien, vous pouvez aller vous balader dans le répertoire `/mnt/tmp/` pour voir le contenu de l'ISO. Une fois que vous avez vérifié que tout est bon, démontez l'ISO du CD :

```
# umount /mnt/tmp
```

Graver le CD

Les options de wodim

Maintenant que l'ISO est créée, vous allez utiliser la commande **wodim** (*write optical disc medium*) pour graver le CD. Les options à connaître et à utiliser sont :

- **dev=/dev/cdrw** où **/dev/cdrw** est un lien symbolique pointant vers le périphérique correspondant à votre graveur IDE ;
- **-data image_ISO.iso** pour préciser que l'on grave à partir de l'image ISO d'un CD ;
- **blank=all** pour effacer complètement le contenu d'un disque réinscriptible ;
- **blank=fast** pour effacer la table des matières d'un disque réinscriptible ;

Lancer la gravure

Pour lancer la gravure, tapez la commande suivante :

```
# wodim -v dev=/dev/cdrw -data backup.iso
```

Copie directe de lecteur à graveur

Pour réaliser une copie directe de lecteur CD à graveur, il faut lancer wodim avec l'option **-isosize** :

```
# wodim -v dev=/dev/cdrw -isosize /dev/cdrom
```

52.3 Les CD audio

Quel programme utiliser ?

Note
L'utilisation de *wodim* pour graver des CD audio induit une pause de deux secondes entre chaque piste lors de la lecture, car il utilise le mode TAO (Track At Once). Pour éliminer cette pause, il faut graver le CD en mode DAO (Disk At Once).

Graver avec wodim

Pour graver un CD audio avec *wodim*, il faut disposer d'un fichier son au format CDR, WAV ou AU pour chaque piste. Si les fichiers sont au format WAV ou AU, chaque fichier doit être en stéréo, 16-bits à 44100 échantillons / seconde.

Convertir au bon format

Pour convertir un MP3 au format CDR, il suffit d'utiliser **mpg321** :

```
% mpg123 --cdr ma_chanson.cdr ma_chanson.mp3
```

Pour extraire une piste d'un CD audio au format CDR, il faut utiliser **cdparanoia** (disponible dans le paquet du même nom) :

```
% cdparanoia -B -p
```

Graver le CD audio

Il suffit alors de taper :

```
# wodim -v -dev=/dev/cdrw -dao -audio track1.cdr track2.cdr ↩
    track3.cdr [etc.]
```

ou

```
# wodim -v -dev=/dev/cdrw -dao -audio track1.wav track2.wav ↩
    track3.wav [etc.]
```

ou

```
# wodim -v -dev=/dev/cdrw -dao -audio track1.au track2.au ↩
    track3.au [etc.]
```

Chapitre 53

Les screens

53.1 Le concept

Le problème à résoudre est le suivant : vous avez lancé sur votre système un programme qui fonctionne en mode console (un client IRC par exemple). Vous voulez vous délogguer en laissant tourner le programme... et pouvoir vous relogguer plus tard (en local ou à distance) et récupérer le programme à l'écran.

Pour cela, il faut lancer le programme dans un **screen**, qui est une sorte écran virtuel que l'on peut détacher et rattacher :

1. vous ouvrez un screen,
2. vous lancez le programme dedans,
3. si vous voulez vous délogguer et laisser tourner le programme, vous détachez le screen,
4. vous pouvez rattacher le screen et donc retrouver le programme depuis n'importe quelle connexion au système (console locale ou accès distant).

53.2 Installer et utiliser

Installer le paquet

```
# aptitude install screen
```

Nous allons prendre l'exemple de 2 scénarios pour expliquer comment ça fonctionne :

1er scénario

1. Depuis un premier ordinateur, ouvrez un screen depuis une console locale en lui donnant un nom. Pour cela, tapez la commande :

   ```
   % screen -S nom_du_screen
   ```

   Un message d'explication apparaît : appuyez sur **Entrée** pour le zapper. Vous avez alors un prompt normal à l'intérieur du screen. Lancez une application qui tourne en mode console (un client IRC par exemple).

2. Quittez le premier ordinateur en laissant le programme tourner et votre console ouverte (on dit que le screen reste attaché). Vous ouvrez une console sur un deuxième ordinateur et vous vous connectez au premier ordinateur (par une connexion SSH par exemple). Pour rattacher le screen, c'est-à-dire retrouver à l'écran le programme que vous avez lancé dans le screen sur le premier ordinateur, tapez la commande :

   ```
   % screen -x nom_du_screen
   ```

   Si vous ne vous souvenez plus du nom que vous aviez donné à votre screen, lancer simplement la commande **screen -x** et vous verrez la liste des screens avec leurs noms associés.
3. Si la taille de votre console sur le deuxième ordinateur n'est pas la même que la taille de votre console sur le premier ordinateur, ce qui se traduit par un programme qui occupe plus ou moins de place que la taille de l'écran : utilisez alors la combinaison de touches Ctrl-a puis **F**, ce qui a pour effet de redimensionner le programme à la taille de votre nouvelle console.
4. Vous voulez quitter le deuxième ordinateur : détachez le screen par la combinaison de touches Ctrl-a puis **d**. Le message suivant apparaît sur la console :

   ```
   [detached]
   ```

   et vous pouvez vous délogguer du deuxième ordinateur.
5. Vous revenez sur le premier ordinateur et vous retrouvez votre console avec le programme qui tourne à l'intérieur. Si vous avez redimensionné le programme sur le deuxième ordinateur, vous devrez le redimensionner à nouveau avec la même combinaison de touches pour le remettre aux dimensions de votre console initiale.
6. Vous voulez quitter le programme qui ne vous sert plus : quittez le programme normalement puis fermez le screen en faisant comme si vous vous délogguiez (combinaison de touches Ctrl-d ou commande **logout**). Le message suivant s'affiche sur la console :

   ```
   [screen is terminating]
   ```

2ème scénario

1. Depuis un premier ordinateur, vous vous connectez à distance sur un deuxième ordinateur. Vous voulez lancer un programme sur ce deuxième ordinateur et pouvoir le récupérer quand vous voulez et depuis n'importe quel ordinateur. Pour cela, lancez le programme dans un screen : pour faire d'une pierre deux coups, c'est à dire ouvrir le screen et lancer le programme en même temps, tapez :

   ```
   % screen -S nom_du_screen ↩
       commande_qui_lance_le_programme
   ```

2. Vous voulez vous délogguer du premier ordinateur : détachez le screen avec la combinaison de touches Ctrl-a puis **d**, déconnectez-vous du deuxième ordinateur puis délogguez-vous du premier ordinateur.

3. Vous voulez retrouver le programme que vous aviez lancé dans le screen : logguez-vous en local sur le deuxième ordinateur ou connectez-vous sur le deuxième ordinateur à distance depuis un autre ordinateur et tapez la commande suivante pour rattacher le screen que vous aviez détaché :

   ```
   % screen -r nom_du_screen
   ```

   Si vous ne vous souvenez plus du nom que vous aviez donné à votre screen, lancez simplement la commande **screen -r** et vous verrez la liste des screens avec leurs noms associés.

   Vous aurez peut-être besoin de redimensionner le programme avec la combinaison de touches Ctrl-a puis **F**.
4. Vous voulez quitter le programme qui ne vous sert plus : quittez le programme normalement et le screen se fermera tout seul car vous aviez ouvert le screen et lancé le programme en même temps. Le message suivant s'affiche sur la console :

   ```
   [screen is terminating]
   ```

Plusieurs fenêtres dans un screen

A l'intérieur d'un screen, vous pouvez avoir une deuxième fenêtre avec un nouveau shell à l'intérieur. Pour cela, utilisez la combinaison de touches Ctrl-a puis **c**. Vous pouvez en ouvrir autant que vous voulez en répétant cette combinaison de touches. Vous pouvez ensuite passer d'une fenêtre à la suivante par la combinaison de touches Ctrl-a puis **n** (*n* comme *Next*) et passer à la fenêtre précédente par la combinaison de touches Ctrl-a puis **p** (*p* comme *Previous*).

Pour fermer une fenêtre, il suffit de fermer le shell qu'elle contient (combinaison de touches Ctrl-d ou commande **logout**). Le fait de fermer la dernière fenêtre restante provoque la fermeture du screen.

Sixième partie

Conclusion et annexes

Si vous avez suivi cette formation de bout en bout, vous devez maintenant pouvoir vous débrouiller sous Debian GNU/Linux et commencer à apprécier la puissance du système de gestion des paquets.

J'espère que nous vous avons donné le goût d'en savoir plus et d'aller plus loin avec Linux... Pour cela, vous pouvez enchaîner avec les annexes de cette formation, qui présentent d'autres programmes, des configurations réseau avancées ainsi que divers trucs et astuces.

Votre avis sur ce document, les difficultés que vous avez rencontrées et les éventuelles erreurs que vous avez constatées nous intéressent beaucoup ! Merci de nous les envoyer à l'adresse formation-debian chez via.ecp.fr.

Annexe A

Apprendre et se tenir au courant

A.1 Apprendre

Internet

– Léa (Linux entre amis)est un site d'aide aux débutants et de documentation sur les systèmes GNU/Linux.
– La section *documentation* du site officiel Debian français rassemble tous les manuels officiels, notamment le manuel d'installation et la référence Debian.
– Le wiki officiel de Debian contient une bonne documentation sur Debian et son utilisation.
– Le site Auto-hébergement regroupe des informations utiles pour l'installation de serveurs personnels, notamment sous Debian.
– Les forums et les articles du site Andesi pourront être d'un grand secours aux débutants.
– L'encyclopédie libre Wikipédia est une très bonne source de documentation en tout genre.
– Le site Absolinux met à votre disposition des tutoriels pour un usage du logiciel libre facile et adapté à vos besoins allant du plus élémentaire au plus complexe.

Magazines

– Planète Linux est un bimensuel très vulgarisé, destiné aux utilisateurs de Linux.
– Linux Pratique est un magazine bimestriel destiné aux utilisateurs des Linux. Chaque numéro est accompagné d'un CD, qui contient le plus souvent une distribution live ou installable.
– Linux Pratique Essentiel, publié tous les deux mois en alternance avec Linux Pratique Essentiel, présente l'actualité des logiciels libres et des environnements de bureaux, ainsi qu'un dossier de découverte d'un thème particulier.
– GNU/Linux Magazine France est un mensuel très technique sur l'administration Linux, les logiciels libres et la programmation avec des outils libres.
– MISC est un magazine bimestriel assez technique sur la sécurité informatique.

Livres

Framabook est une collection de livres libres sélectionnés par Framasoft.

Les livres des éditions Eyrolles sont très connus et réputés sérieux dans la communauté du logiciel libre. Notons en particulier :

– Le livre Cahiers de l'admin - Debian GNU/Linux de Raphaël Hertzog, Christophe Le Bars et Roland Mas aux éditions Eyrolles.

Les LUGs

LUG signifie Linux User Group. Ce sont des associations locales qui regoupent les fans de Linux et de l'informatique libre : rien de tel pour rencontrer des passionnés de Linux prêt à vous aider ! Linux.org recense les LUGs classés par pays ; une liste des LUGs français est disponible sur la Wikipédia, et une carte sur LinuxFrench.Net. De nombreux LUGs organisent des rencontres régulières, qui sont référencés sur le site FirstJeudi.org.

A.2 Trouver de l'aide

Forums Web

– Léa (Linux entre amis) comporte un forum d'aide aux débutants.
– Le site debian-fr.org propose des forums de discussion dédiés à Debian.
– Le site Andesi comporte également de nombreux forums liés à Debian.
– Sur Absolinux vous trouverez des forums d'entraide et de discussions autour de Linux et *BSD.

Forums USENET

Il existe plusieurs forums USENET dédiés aux systèmes GNU/Linux, en langue anglaise ou française, par exemple *fr.comp.os.linux.configuration*

Canaux de discussion

Il est souvent possible d'obtenir une aide rapide sur les réseaux IRC ou Jabber :

– le réseau Freenode héberge la majorité des canaux de discussion liés au logiciel libres, notamment les canaux : #linuxfr #absolinux ;
– le réseau OFTC héberge entre autre les canaux de discussion liés à Debian : #debian, #debian-fr, ainsi que notre canal #formation-debian.

A.3 Se tenir au courant

Voilà une liste de sites Webs traitant de l'actualité de Debian, Gnu, Linux et des logiciels libres :

– Les actualités Debian donnent les dernières informations officielles de Debian (vous pouvez aussi vous abonner à une liste de distribution pour les recevoir par courrier électronique),
– Debian News traite de l'actualité Debian en anglais,
– Planet Debian regroupe les derniers articles de sites personnels sur Debian,
– LinuxFr est *le* site de l'actualité Linux et des logiciels libres en français,
– Slashdot (ou **/.**) est son équivalent anglais, à beaucoup plus forte audience,
– Linux Today un site anglais d'actualité généraliste sur Linux,

- FreshMeat est un site anglais qui rassemble toutes les informations sur les logiciels Unix et multiplateformes. Très utile pour se tenir au courant des sorties de nouvelles versions ou chercher les logiciels libres existant dans un certain domaine.

Annexe B

En cas de blocage...

B.1 Méthode

Tout d'abord, il faut perdre le réflexe de rebooter la machine à chaque fois qu'un problème ou un blocage intervient. Si cette méthode à fait ses preuves avec Windows, ce n'est pas une bonne méthode pour Linux. La méthode à adopter dépend de ce qui est bloqué, comme expliqué ci-dessous.

B.2 Blocage d'un service

Quand un service tel qu'un serveur Web, un serveur mail, etc... est bloqué ou ne marche plus comme il devrait, la première chose à faire est de consulter les logs du service concerné. Le service a parfois ses propres fichiers de logs (par exemple, le répertoire `/var/log/apache/` regroupe tous les fichiers de logs d'Apache) ou il écrit dans le fichier de log générique `/var/log/daemon.log`. Les messages écrits dans les fichiers de logs ne sont pas toujours facile à comprendre, mais c'est LE moyen de diagnostiquer un problème avec un service.

Pour suivre en direct les messages qui s'écrivent dans un fichier de log (par exemple le fichier `/var/log/daemon.log`) :

```
% tail -f /var/log/daemon.log
```

Une des solutions peut être de redémarrer le service. Par exemple, pour redémarrer Apache :

```
# /etc/init.d/apache restart
Reloading apache modules.
```

B.3 Blocage à cause d'un processus

Si le processus peut être identifié

Il peut arriver qu'un programme lançé par un utilisateur plante et se mette à utiliser toutes les ressources processeur disponibles. Dans ce cas, il faut identifier le PID du

programme en cause avec le programme **top** et tuer le processus en question avec la commande **kill**, dont l'utilisation était expliquée au Chapitre 14.

Dans certains cas, une solution plus radicale est nécessaire : tuer d'un seul coup tous les processus appartenant à un utilisateur. Pour cela, il faut utiliser le programme *slay* qui n'est pas installé par défaut :

```
# aptitude install slay
```

Pour tuer tous les processus de l'utilisateur *toto* :

```
# slay toto
```

Quand on ne peut plus accéder à une console locale

Dans certains cas, comme par exemple quand le serveur graphique se bloque, on ne peut plus accéder à une console. Si l'ordinateur est en réseau, il faut avoir le réflexe de s'y connecter à distance par *ssh* pour tuer les processus bloquants.

Les touches magiques

En cas de blocage grave du système, et si toutes les solutions précédentes n'ont pas fonctionné, il existe une ultime méthode : *les touches magiques*.

Dans un premier temps, essayez d'utiliser les combinaisons de touches susceptibles de tuer le processus bloquant :

- Alt-Impr. écran-k tue tous les processus de la console courante,
- Alt-Impr. écran-e envoie le signal *TERM* à tous les processus du système, sauf le processus père *init*,
- Alt-Impr. écran-i envoie le signal *KILL* (plus autoritaire que le précédent) à tous les processus du système, sauf *init*.

Si ces combinaisons de touches ne permettent pas de récupérer une console sur le système, il faut alors se résoudre à rebooter le système en utilisant successivement les combinaisons de touches suivantes à quelques secondes d'écart :

1. Ctrl-Alt-Impr. écran-r (rend le contrôle du clavier au noyau, souvent nécessaire lorsqu'on utilise le système graphique, qui prend la main dessus)
2. Alt-Impr. écran-s (synchronise les systèmes de fichiers)
3. Alt-Impr. écran-u (démonte les systèmes de fichiers)
4. Alt-Impr. écran-b (reboote le système)

Note

En réalité, la touche à utiliser n'est pas **Impr. écran** mais **Syst** (parfois également appelée **SysRq**) qui est parfois située ailleurs sur le clavier. Sur certains ordinateurs portables il peut être nécessaire d'utiliser la touche **fn**.

ASTUCE

Une séquence complète pour un redémarrage d'urgence consiste à rendre le contrôle du clavier au noyau, terminer les processus, tuer ceux qui restent, synchroniser les systèmes de fichiers, les démonter, puis redémarrer : les touches pour effectuer cette séquence correspondent aux initiales des mots de la phrase mnémotechnique « revenir en Islande sur un bateau ».

Vous pouvez également afficher une liste des combinaisons de touches magiques disponibles en utilisant la combinaison Alt-Impr. écran-h.

B.4 Blocage au boot ou problème avec le boot loader

Vous avez mal configuré votre boot loader et vous n'arrivez même plus à démarrer ? Vous avez réinstallé Windows et il a écrasé LILO ou Grub ? Vous avez mal configuré GDM et il bloque l'ordinateur dès le lancement ? Rassurez-vous, vous n'aurez pas à tout réinstaller ; suivez la procédure suivante pour vous en sortir :

1. Suivez toutes les instructions du Chapitre 7 comme si vous recommenceriez la procédure d'installation.
2. Une fois que vous avez choisi votre clavier, passez sur la deuxième console avec la combinaison de touches habituelle et appuyez sur **Entrée** pour activer la console.
3. Montez votre partition Linux racine :
   – si elle est sur un disque IDE :

```
# mkdir /mnt/root
# mount /dev/ide/host0/bus0/targetX/lun0/partY /mnt/ ↩
    root
```

où `/dev/ide/host0/bus0/targetX/lun0/partY` représente le disque dur où se trouve votre partition racine en « notation » *devfs*. Voilà quelques exemples de correspondance entre la notation traditionnelle et la notation devfs qui vous permettront de comprendre comment ça marche :

Notation traditionnelle	**Notation devfs**
/dev/hda1	/dev/ide/host0/bus0/target0/lun0/part1
/dev/hda2	/dev/ide/host0/bus0/target0/lun0/part2
/dev/hdb2	/dev/ide/host0/bus0/target1/lun0/part2
/dev/hdc5	/dev/ide/host0/bus0/target2/lun0/part5

TABLE B.1 –

   – si elle est sur un disque SATA ou SCSI :

```
# mkdir /mnt/root
# mount /dev/scsi/host0/bus0/targetX/lun0/partY /mnt/ ↩
    root
```

où *X* désigne le numéro de votre disque SATA ou SCSI et *Y* désigne le numéro de votre partition Linux racine.

4. Demandez au système que le répertoire racine devienne `/mnt/root/` :

```
# chroot /mnt/root
```

Vous vous retrouvez alors dans votre système de fichiers habituel.

5. Pour plus de confort, lancez votre shell habituel :

```
# zsh
```

6. Vous pouvez alors faire toutes les manipulations nécessaires à la réparation, par exemple éditer `/boot/grub/menu.lst` avec *vim*.
7. Une fois que la réparation est terminée, délogguez-vous deux fois pour retrouver votre prompt initial et rebootez le système avec la commande **reboot**.

Annexe C

Utilisation avancée d’aptitude

Dans la Partie II, vous avez pu découvrir une utilisation simple d’aptitude :

```
% aptitude show paquet
# aptitude install paquet
% aptitude remove paquet
% aptitude purge paquet
% aptitude update paquet
% aptitude safe-upgrade paquet
% aptitude full-upgrade paquet
```

aptitude a été conçu comme un outil de gestion des paquets extrêmement puissant : nous allons vous faire découvrir quelques-unes de ses possibilités.

C.1 Utilisation en ligne de commande

Gérer les paquets installés automatiquement

Lorsque vous installez un logiciel, plusieurs bibliothèques sont souvent installées par dépendances. Ainsi, l’installation de *xmoto* entraîne celle de *libsdl-ttf2.0-0*, par exemple. aptitude note alors que le paquet *libsdl-ttf2.0-0* a été installé automatiquement. Lorsque vous désinstallez des paquets, aptitude désinstalle également les paquets ainsi marqués qui ne sont plus nécessaires.

Vous pouvez demander vous-même à aptitude de marquer un paquet comme installé automatiquement, ou installé manuellement, avec les commandes suivantes :

```
# aptitude markauto paquet
# aptitude unmarkauto paquet
```

Rechercher des paquets

Pour rechercher des paquets, on utilise habituellement la commande **apt-cache**, mais aptitude permet également d’effectuer des recherches, de façon bien plus souple.

Principe

Pour rechercher un paquet, on utilise la commande **aptitude search**, suivie d'un ou plusieurs arguments appelés *motifs*. aptitude recherche alors les paquets qui correspondent à *au moins l'un de ces motifs*. Les motifs peuvent être composés de plusieurs termes de recherche, qui alors être tous vérifiés pour qu'un paquet corresponde à ce motif. Un exemple vous permettra d'y voir plus clair :

– pour rechercher tous les paquets dont le nom contient *gnome*, on utilise deux motifs de recherche :

```
% aptitude search kde gnome
```

– pour rechercher tous les paquets dont le nom contient *gnome* et *kde*, on utilise un motif de recherche composé de deux termes :

```
% aptitude search 'kde gnome'
```

– pour rechercher tous les paquets dont le nom contient soit les mots *gnome* et *kde*, soit le le mot *xfce*, on utilise deux motifs de recherche, dont l'un est composé de deux termes :

```
% aptitude search 'gnome kde' xfce
```

Motifs de recherche

aptitude recherche normalement les termes demandés dans les titres des paquets, mais il existe plusieurs motifs de recherche :

`?name(mot)` ou `~nmot` les paquets dont le nom contient *mot* ;

`?description(mot)` ou `~dmot` les paquets dont la description contient *mot* ;

`?installed` ou `~i` les paquets installés ;

`?automatic` ou `~M` les paquets installés non pas manuellement, mais automatiquement, comme dépendance d'autres paquets ;

`?not(terme)` ou `!terme` les paquets qui ne correspondent pas au *terme* de recherche (négation) ;

`?and(terme1, terme2)` ou `terme1 terme2` les paquets qui correspondent au *terme1* de recherche et au *terme2* ;

`?or(terme1, terme2)` ou `terme1 | terme2` les paquets qui correspondent au *terme1* de recherche ou au *terme2*.

Affichage du résultat

Supposons que je veuille connaître la liste des paquets que j'ai moi-même installés (c'est à dire qu'ils n'ont pas été installés par dépendance), et dont la description contient le mot *docbook* et soit le mot *pdf*, soit le mot *html* :

```
% aptitude search '~i !~M (~ddocbook | ~dhtml)'
i   dblatex               - Produces DVI, PostScript, PDF  ↩
    documents from DocBook sources
i   docbook-xsl           - stylesheets for processing  ↩
    DocBook XML files to various output
i   docbook-xsl-doc-html    - stylesheets for processing  ↩
    DocBook XML files (HTML documentation)
```

aptitude affiche alors une liste, dont chaque ligne comprend :

– l'état du paquet :
  - **`i`** pour les paquets installés,
  - **`c`** pour les paquets désinstallés mais dont les fichiers de configuration ont étés conservés (c'est à dire non purgés),
  - **`p`** pour les paquets dont aucune trace n'est présente sur le système ;
– le nom du paquet ;
– la description courte du paquet.

C.2 Utilisation de l'interface semi-graphique

aptitude dispose également d'une interface semi-graphique de type *ncurses*, qui offre plusieurs avantages :

– elle est interactive, ce qui permet par exemple d'affiner progressivement une recherche ;
– elle affiche des informations utiles au fur et à mesure ;
– elle peut être lancée en tant que simple utilisateur, et permet de devenir *root* lorsqu'il y en a besoin, par exemple pour installer un paquet.

Lancement

Pour utiliser l'interface semi-graphique d'aptitude, utilisez simplement la commande **aptitude** sans argument :

```
% aptitude
```

```
 Actions  Annuler  Paquet  Solutions  Rechercher  Options  Vues  Aide
C-T: Menu  ?: Aide  q: Quitter  u: MAJ  g: Téléch./Install./Suppr. Paqts
aptitude 0.4.11.8
--- Nouveaux paquets (2411)
--- Paquets installés (1428)
--- Paquets non installés (21031)
--- Paquets obsolètes ou créés localement (2)
--- Paquets virtuels (2579)
--- Tâches (715)

Ces paquets ont été ajoutés à Debian depuis la dernière fois que vous
avez nettoyé la liste des paquets « nouveaux » (Choisir « oublier les
nouveaux paquets » depuis le menu Actions pour vider cette liste).

Ce groupe contient 2411 paquets.
```

FIGURE C.1 – Fenêtre principale d'aptitude

Après un court temps de chargement, vous arrivez dans la fenêtre principale d'aptitude, Figure C.1. Vous pouvez alors accéder aux menus avec la touche **F10** ou la combinaison Ctrl-T. Appuyez sur **Entrée** pour confirmer une commande, ou sur **Échap** pour sortir des menus.

Navigation dans les paquets

Dans la fenêtre principale, vous trouverez plusieurs rubriques : *Paquets installés*, *Paquets non-installés*... Pour ouvrir ou refermer une rubrique, allez sur celle-ci avec les flèches, puis appuyez sur **Entrée**. ne fois qu'une rubrique est ouverte, vous avez accès à des groupes de programmes, que vous pouvez ouvrir et refermer de la même façon.

Lorsqu'un groupe de programmes est ouvert, une liste des paquets disponibles s'affiche. La plupart du temps, celle-ci sera plus grande que la fenêtre d'aptitude, et vous pourrez vous déplacer dedans en utilisant les flèches du clavier.

Décrire et installer un paquet

Si vous voulez en savoir plus sur le paquet où vous vous trouvez, appuyez sur **Entrée** pour ouvrir une nouvelle fenêtre où seront affichés tous les détails tels que les dépendances, les conflits, etc. Appuyez sur **q** pour revenir à la fenêtre principale.

Pour installer ou désinstaller in paquet, ouvrez le menu avec **F10**, pour déplacez-vous sur le menu *Paquet* et sélectionnez l'action correspondante : *Installer*, *Supprimer*, *Purger*... Vous pouvez également utiliser les raccourcis tels que **+**, **-** proposés dans ce menu.

Lorsque vous avez sélectionné les paquets à installer ou à désinstaller, appuyez sur **g** pour effectuer les actions demandées. aptitude affiche alors un récapitulatif des actions à effectuer (Figure C.2 : appuyer encore une fois sur **g** pour valider.

```
 Actions  Annuler  Paquet  Solutions  Rechercher  Options  Vues  Aide
C-T: Menu  ?: Aide  q: Quitter  u: MAJ  g: Téléch./Install./Suppr. Paqts
                   Paquets                               Aperçu
aptitude 0.4.11.8                     Libérera 37,4Mo d'espace disqu 1236ko à téléch
--\ Paquets qui vont être supprimés parce qu'ils ne sont plus utilisés (5)
idA  gnome-games-data                                 -35,5MB 1:2.22.3-2 1:2.22.3-2
idA  gnuchess                                         -279kB  5.07-4.1   5.07-4.1
idA  libggz2                                          -176kB  0.0.14.1-1 0.0.14.1-1
idA  libggzcore9                                      -831kB  0.0.14.1-1 0.0.14.1-1
idA  libggzmod4                                       -131kB  0.0.14.1-1 0.0.14.1-1
--\ Paquets ajoutés automatiquement pour satisfaire des dépendances (3)
piA  libgsasl7                                        +618kB  <aucun>    0.2.26-1
piA  libmailutils1                                    +2073kB <aucun>    1:1.2+dfsg
piA  libntlm0                                         +81,9kB <aucun>    0.3.13-1
--\ Paquets qui vont être effacés parce qu'ils ont des dépendances non satisfaites (1)
id   bsd-mailx                                        -307kB  8.1.2-0.20 8.1.2-0.20
--\ Paquets à installer (1)
pi   mailutils                                        +856kB  <aucun>    1:1.2+dfsg
--\ Paquets à enlever (2)
idA  gnome-cards-data                                 -878kB  1:2.22.3-2 1:2.22.3-2
id   gnome-games                                      -2886kB 1:2.22.3-2 1:2.22.3-2
--- Paquets suggérés par d'autres paquets (1)

Ces paquets vont être supprimés parce qu'ils ont été installés automatiquement pour
répondre à des dépendances et l'action en cours conduira à ce que plus aucun des
paquets ne les déclarera comme une dépendance « importante ».

Si vous sélectionnez un paquet, une explication de son état courant apparaîtra à cet
endroit.

Voir ou modifier les actions qui seront effectuées
```

FIGURE C.2 – Récapitulatif des actions à effectuer

AVERTISSEMENT

Si vous avez lancé aptitude en tant que sélectionné un programme pour l'installer ou le désinstaller, aptitude ouvrira alors une fenêtre rouge pour vous demander votre mot de passe administrateur afin de devenir *root* avant de démarrer l'installation. Comme expliqué à la Partie I, seul *root* peut effectuer des tâches d'administration, ce qui évite que n'importe qui installe n'importe quoi sur votre ordinateur.

Pour quitter aptitude, appuyez sur **q**.

Annexe D

Compléments sur la gestion des paquets Debian

D.1 Tout savoir sur l'utilisation d'APT

Pour tout savoir sur l'utilisation d'APT, je vous invite à installer le manuel d'aptitude, disponible dans le paquet *aptitude-doc-fr*, puis à le consulter directement sur votre système.

D.2 dpkg-deb

Dpkg a un grand frère, **dpkg-deb** qui sert à construire et manipuler des paquets Debian, mais pas à les installer ou désinstaller.

Les commandes qui peuvent vous être utiles sont les suivantes :

- **dpkg-deb -I package_0.1_i386.deb** : affiche les en-têtes du paquet.
- **dpkg-deb -c package_0.1_i386.deb** : affiche les noms des fichiers contenus dans le paquet.
- **dpkg-deb -X package_0.1_i386.deb répertoire** : extrait les fichiers contenus dans le paquet dans le répertoire spécifié.

D.3 Apt-file

Apt-file est un programme qui vient compléter les commandes *dpkg -S* et *dpkg -L* qui se limitent dans leurs résultats aux paquets installés.

Pour utiliser *apt-file*, il faut d'abord installer le paquet du même nom, puis mettre à jour sa base de donnée propre contenant la liste des fichiers de tous les paquets :

```
# aptitude install apt-file
# apt-file update
```

Puis on peut avoir la liste des fichiers contenus dans un paquet même si celui-ci n'est pas installé (contrairement à la commande *dpkg -L*) :

```
% apt-file list nom_du_paquet
```

On peut également savoir à quel paquet appartient un certain fichier, même si le paquet (et donc le fichier) n'est pas installé (contrairement à la commande *dpkg -S*) :

```
% apt-file search nom_du_fichier
```

D.4 Installer un RPM

RPM signifie *Red Hat Packet Manager* : c'est un format de paquet concurrent de celui de Debian, et il est utilisé par de nombreuses distributions (Red Hat, Mandriva, SUSE, etc...).

Pour certaines applications, on ne trouve que des paquets au format RPM. Dans ce cas, il faut utiliser **alien** pour convertir un paquet du format RPM au format Debian.

Commencez par installer *alien* :

```
# aptitude install alien
```

Puis convertissez le paquet :

```
# alien -d nom_du_paquet.rpm
```

Enfin, installez le paquet Debian généré :

```
# dpkg -i nom_du_paquet.deb
```

Annexe E

Faire marcher une imprimante

E.1 Préliminaires

Mon imprimante est-elle prise en charge ?

AVERTISSEMENT

Utiliser une imprimante sous GNU/Linux n'est pas difficile en soi… à condition qu'elle soit prise en charge ! Debian utilise le même système d'impression que MacOS X, et toutes les imprimantes « compatibles MacOS » devraient donc fonctionner sans problèmes. Certains constructeurs comme HP et Epson collaborent avec la communauté du logiciel libre, et leurs imprimantes sont très bien prises en charge. D'autres, comme Canon ou Lexmark font preuve de mauvaise volonté, et leurs imprimantes, qui sont pourtant parfois excellentes, peuvent être inutilisables.

Tout d'abord, il faut partir aux renseignements pour savoir si votre imprimante est prise en charge. Pour cela, rendez-vous sur le site OpenPrinting.org et cliquez sur *Printers* : sélectionnez la marque et le modèle de votre imprimante, et vous affiche alors une page qui vous informe sur la qualité de la prise en charge de votre imprimante. Si cette page vous dit que votre imprimante est un *Paperweight* avec un pingouin barré comme logo, alors ce n'est pas la peine d'aller plus loin. Dans les autres cas, si vous avez un, deux ou trois pingouins, c'est que vous devez pouvoir faire quelque chose de votre imprimante sous Linux : continuez à lire ce chapitre !

Note

Plusieurs fabriquants distribuent des imprimantes affichant un support pour la forme, et distribuent de prétendus « pilotes » compliqués à installer et à utiliser. D'une façon générale, il vaut mieux utiliser des pilotes non officiels, qui sont de bien meilleure qualité et plus faciles à utiliser que ceux que vous pourrez trouver sur les sites et les disques d'installation des imprimantes.

Un peu de théorie

Quelques connaissances théoriques sont nécessaires pour comprendre les différents formats et les différentes couches logicielles impliquée dans la chaîne d'impression en

fonction des capacités de votre imprimante. La lecture de ce document (en anglais) vous permettra de beaucoup mieux comprendre la suite.

Noter le pilote recommandé

Maintenant que vous avez les bases théories requises, retournez dans *Printer listing* sur le site LinuxPrinting.org, sélectionnez à nouveau votre imprimante et notez le pilote recommandé pour faire marcher votre imprimante sous Linux :

- si le pilote recommandé est *Postscript*, installez le paquet *openprinting-ppds* ;
- si le pilote recommandé est *Gutenprint*, installez le paquet *cups-driver-gutenprint* ;
- si le pilote recommandé est *HPLIP*, installez le paquet *hplip* ;
- si le pilote recommandé est un fichier PPD spécifique, téléchargez-le pour vous en servir par la suite.

E.2 Installer et configurer CUPS

Installer CUPS

Installez les paquets de CUPS :

```
# aptitude install cups cups-client cups-bsd
```

Configurer CUPS

La configuration de CUPS se fait via une interface Web, qui, par défaut, n'est accessible qu'en local. Ouvrez donc votre navigateur web favori (si vous n'avez pas de serveur graphique, vous pouvez utiliser un navigateur web en console comme *w3m*) et tapez l'URL http://localhost:631/admin/.

Note

Lorsque vous effectuez des modifications à l'aide de cette interface, avant de les enregistrer, celle-ci vous demande de vous identifier : saisissez alors *root* comme login et votre mot de passe root (comme vous accédez à l'interface web en local sur la machine, votre mot de passe root ne sera donc pas transmis en clair à l'extérieur).

Note

Si vous voulez qu'un utilisateur qui ne connaît pas le mot de passe root puisse administrer CUPS par l'interface web, ajoutez-le au groupe *lpadmin* ; il pourra alors y accéder avec son login et son mot de passe.

Vous arrivez alors dans l'interface d'administration de CUPS :

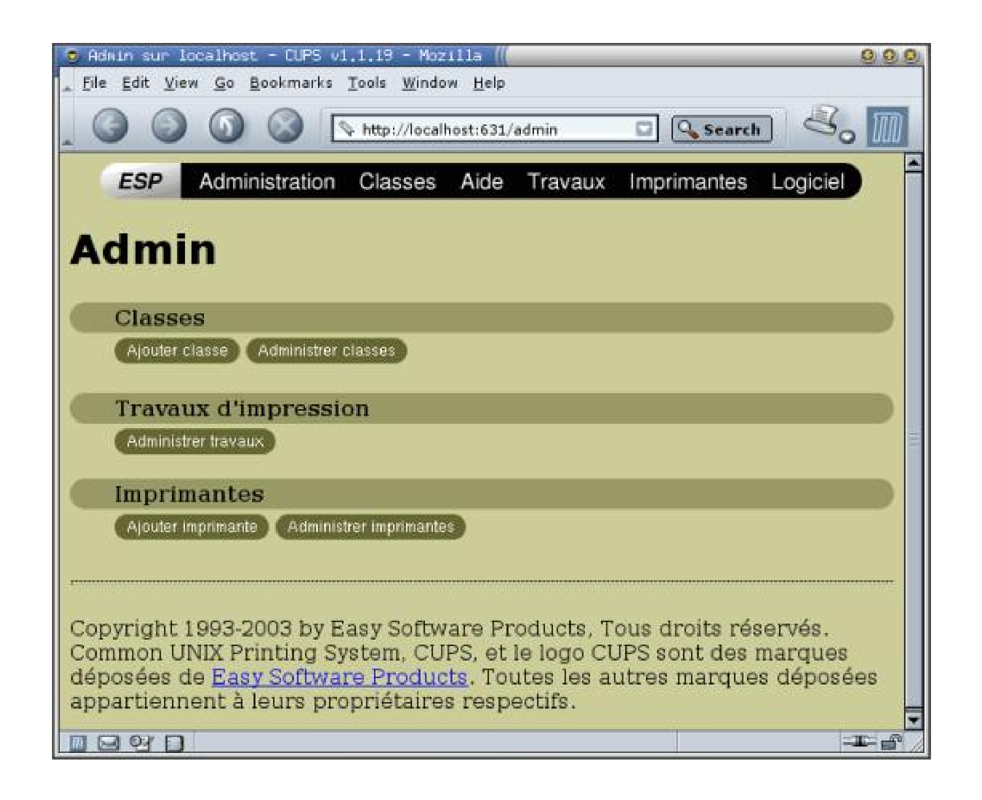


FIGURE E.1 – Interface d'administration de CUPS

Cliquez sur *Ajouter une imprimante* et suivez l'assistant :

1. donnez un nom (sans espace) à votre imprimante (et éventuellement un emplacement et une description) ;
2. sélectionnez le périphérique d'accès à l'imprimante : si c'est une imprimante sur port parallèle, sélectionnez *Parallel port #1*, si c'est une imprimante sur port USB, sélectionnez *USB printer #1* ;
3. si le pilote recommandé était un fichier PPD spécifique, sélectionnez-le à l'aide du bouton adéquat ;
4. sinon, sélectionnez la marque de votre imprimante ;
5. sélectionnez le modèle de votre imprimante : si le nom exact n'apparaît pas, rappelez-vous du nom du pilote qui était recommandé par OpenPrinting.org.

Il vous informe alors que l'imprimante a été ajoutée avec succès. Cliquez sur le nom de l'imprimante pour aller directement à la page de statut de celle-ci :

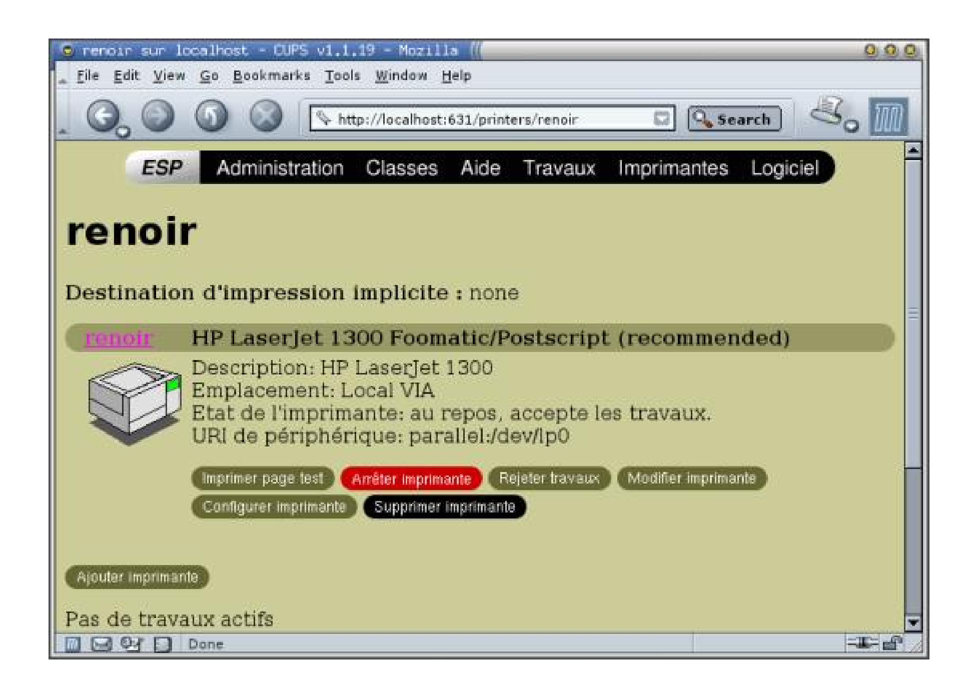


FIGURE E.2 – Page de status de l'imprimante

Si le champ d'état de l'imprimante indique qu'elle est à l'arrêt, cliquez sur *Démarrer l'imprimante*. Ensuite, cliquez sur *Configurer l'imprimante* pour régler un certain nombre de paramètres (taille du papier, résolution...). Enfin, cliquez sur *Imprimer page de test* pour voir si vous êtes capable de faire cracher quelque chose à votre imprimante sous Linux !

E.3 Imprimer !

Les commandes de base

Tout d'abord, il faut connaître quelques commandes de base :

- **lpq** affiche l'état de l'imprimante et la queue d'impression ;
- **lprm 12** supprime le *job* numéro 12 de la queue d'impression ;
- **lp rapport.ps** ajoute le fichier PostScript `rapport.ps` à la queue d'impression de l'imprimante (si plusieurs imprimante sont configuées dans CUPS, il faut ajouter **-d nom_de_l'imprimante** dans la commande) ;
- **lp -n 3 -P 2,12-42,72 -d renoir rapport.ps** ajoute à la queue d'impression de l'imprimante *renoir* l'impression en trois exemplaires des pages 2 et 72 ainsi que de toutes les pages entre 12 et 42 du fichier `rapport.ps`.

Imprimer depuis une application

La plupart des applications graphiques proposent un menu d'impression comme sous Windows... donc vous ne devriez pas être trop dérouté !

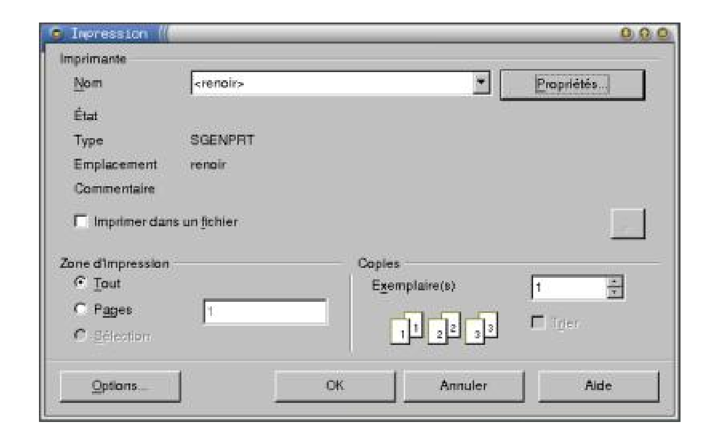


FIGURE E.3 – Menu d'impression d'OpenOffice

E.4 Transformer l'ordinateur en serveur d'impression

Maintenant que l'impression en local marche, il est facile de transformer l'ordinateur en serveur d'impression ; les autres ordinateurs du réseau pourront alors se servir de l'imprimante à distance !

Configuration du serveur d'impression

Sur l'ordinateur auquel est connecté l'imprimante (désormais appelé *serveur d'impression*), éditez en root le fichier `/etc/cups/cupsd.conf` et positionnez-vous au niveau des lignes suivantes :

```
<Location />
Order Deny,Allow
Deny From All
Allow From 127.0.0.1
</Location>
```

Modifiez le contenu de ce paragraphe pour autoriser d'autres ordinateurs du réseau à utiliser le serveur Cups de l'ordinateur. Par exemple, pour que tous les ordinateurs ayant des IPs comprises entre 192.168.0.1 et 192.168.0.255 puissent utiliser le serveur d'impression, modifiez le paragraphe pour qu'il contienne :

```
<Location />
Order Deny,Allow
Deny From All
Allow From 192.168.0.0/255.255.255.0
</Location>
```

Ensuite, dites à Cups de relire son fichier de configuration :

```
# /etc/init.d/cupsys reload
Reloading CUPSys: cupsd.
```

Configuration des clients sous Debian GNU/Linux

Sur chaque ordinateur du réseau sous Debian susceptible d'utiliser le serveur d'impression, exécutez les instructions ci-dessous.

Tout d'abord, installez le programme client de Cups :

```
# aptitude install cupsys-client
```

Ensuite, éditez le fichier `/etc/cups/client.conf` et décommentez la ligne commençant par *ServerName*. Sur cette ligne, vous devez alors préciser l'adresse IP ou le nom DNS du serveur d'impression.

Par exemple, si votre serveur d'impression a l'adresse IP *192.168.0.42*, le fichier `/etc/cups/client.conf` devra contenir :

```
ServerName 192.168.0.42
```

Vous pouvez désormais imprimer depuis le poste client comme si l'imprimante était connectée directement à l'ordinateur : la commande est la même !

Annexe F

La souris en console

Vous utilisez Linux principalement en console et vous voulez pourvoir utiliser votre souris en console ? Cela est possible grâce au programme *GPM* (General Purpose Mouse interface). Ce programme permet notamment de faire du copier coller dans la même console, ou d'une console à une autre.

F.1 Installer et configurer GPM

Commencez par fermer votre serveur graphique si vous en avez un. Ensuite, installez le paquet de GPM :

```
# aptitude install gpm
```

Dès l'installation, il vous propose de configurer GPM. Je vous conseille de répondre *Non* à la question *Do you want to change anything ?* et de configurer GPM à la main par la suite.

Pour modifier la configuration de GPM, éditez son fichier de configuration `/etc/gpm.conf`. Les lignes non commentées sont les suivantes :

```
device=/dev/psaux
responsiveness=
repeat_type=ms3
type=autops2
append=""
sample_rate=
```

Le paramètre *device* doit contenir le device correspondant à ta souris :

- `/dev/psaux` pour une souris sur le port PS/2 (les trackpoints et les touchpads des ordinateurs portables sont généralement raccordés à ce port),
- `/dev/input/mice` pour une souris sur port USB,
- `/dev/ttyS0` pour une souris sur le premier port série,
- `/dev/ttyS1` pour une souris sur le second port série,

Le paramètre *type* définit le format des données envoyées par la souris et lu par le système à travers le device. Indiquez comme format :

- `ps2` si vous avez une souris de base,
- `imps2` si vous avez une souris un peu plus perfectionnée,
- ... il existe plein d'autres formats. Pour avoir la liste complète des protocoles avec le type de souris qui les utilise, lancez la commande **gpm -t help**.

Une fois que vous avez personnalisé les paramètres *device* et *type*, enregistrez le fichier et quittez votre éditeur de texte. Il faut alors relancer GPM pour qu'il tienne compte du changement dans le fichier de configuration :

```
# /etc/init.d/gpm restart
Stopping mouse interface server: gpm.
Starting mouse interface server: gpm.
```

Si vous obtenez un curseur qui bouge... c'est gagné ! Sinon, ré-éditez le fichier de configuration, modifiez le paramètre *type*, enregistrez, et relancez GPM... et ainsi de suite jusqu'à ce que ça marche !

F.2 Utiliser GPM

Avec la souris, vous pouvez maintenant *sélectionner* du texte avec le bouton gauche. Pour sélectionner un mot, vous pouvez aussi vous placer sur ce mot et double-cliquer. Un triple-clic sélectionne la ligne entière.

Vous pouvez alors *coller* le texte que vous avez sélectionné à l'endroit où est votre curseur avec un clic droit.

Annexe G

Outils Windows pour Linuxiens

De plus en plus de programmes Linux sont portés sous Windows... ce qui permet aux Linuxiens de retrouver leurs logiciels favoris quand ils sont sous Windows ! Certains programmes ont aussi été développés spécialement pour permettre une interopérabilité Linux/Windows.

G.1 GNUwin

Le projet GNUWin regroupe un grand nombre de *logiciels libres* fonctionnant sous Windows. Dans la liste des logiciels proposés, nous utilisons notamment PuTTY :

PuTTY

PuTTY est un client Telnet et SSH.

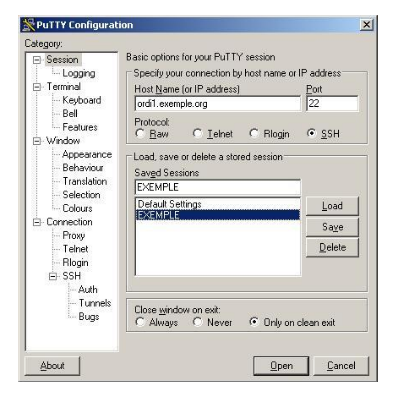


FIGURE G.1 – PuTTY

Cygwin

Cygwin est un environnement Unix complet pour Windows. Il permet de se servir de très nombreux programmes Unix tel qu'un serveur graphique, un certain nombre

d'applications graphiques et la plupart des outils en ligne de commande (comme *cvs* par exemple).

Installer Cygwin

Allez sur www.cygwin.com et cliquez sur l'icône *Install Cygwin now* qui se trouve en haut à droite de la page. Téléchargez le fichier `setup.exe` et exécutez-le. Le fichier contient le programme d'installation, mais pas Cygwin en lui-même.

La procédure d'installation démarre alors :

1. Une fois passé l'écran d'accueil, sélectionnez *Install from Internet*.
2. Sélectionnez le répertoire d'installation.
3. Sélectionnez un répertoire dans lequel il va écrire les fichiers qu'il va télécharger.
4. Si vous devez passer par un proxy pour accéder à Internet, entrez ses paramètres. Sinon, sélectionnez *Direct Connection*.
5. Sélectionnez un miroir dans la liste. Si vous êtes connecté au réseau VIA, entrez l'adresse *ftp://ftp.via.ecp.fr/pub/cygwin/* et cliquez sur *Add*.
6. Ensuite vient l'étape de sélection des paquets. Sélectionnez deux paquets supplémentaires par rapport à la configuration par défaut :
   – *openssh* dans la section *Net*,
   – *XFree86-base* dans la section *XFree86*.
7. Il va ensuite télécharger les paquets sélectionnés et les installer.
8. Après la dernière étape, il lance les scripts de configuration-après-installation et ajoute l'icône *Cygwin* sur le bureau.

Utiliser Cygwin

Double-cliquez sur l'icône *Cygwin* ; une console apparaît :

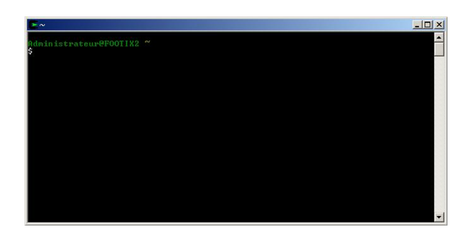


FIGURE G.2 – La console Cygwin

Vous pouvez alors vous servir de tous les outils Unix disponibles avec Cygwin... comme si vous étiez sous Linux !

Si vous voulez lancer un serveur X, commencez par éditez le fichier `/usr/X11R6/bin/startxwin.bat` et rajoutez à la fin du fichier la ligne suivante :

```
run setxkbmap -layout fr
```

ce qui vous permettra d'avoir un clavier français sous X. Vous pouvez maintenant lancer le serveur graphique :

```
Administrateur@CLIENT
$ startxwin.bat
```

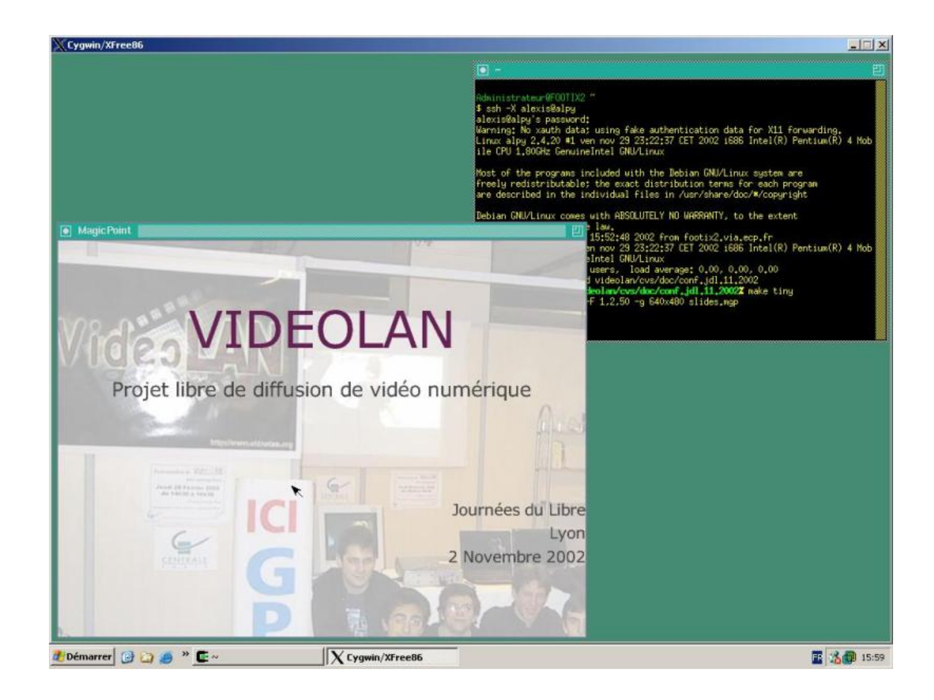


FIGURE G.3 – Serveur X de Cygwin

Depuis la console Cygwin, vous pouvez également lancer un serveur graphique vers une autre machine, comme si vous étiez sur une console Linux :

```
Administrateur@CLIENT
$ X -query serveur.exemple.org
```

Cela permet de faire de l'export display de Linux vers Windows, comme expliqué au Chapitre 42.

G.2 WinSCP

WinSCP est un client SFTP pour Windows sous licence GPL. Il marche exactement comme un client FTP, mais il utilise le protocole SSH pour sécuriser les transferts de mots de passes et de fichiers.

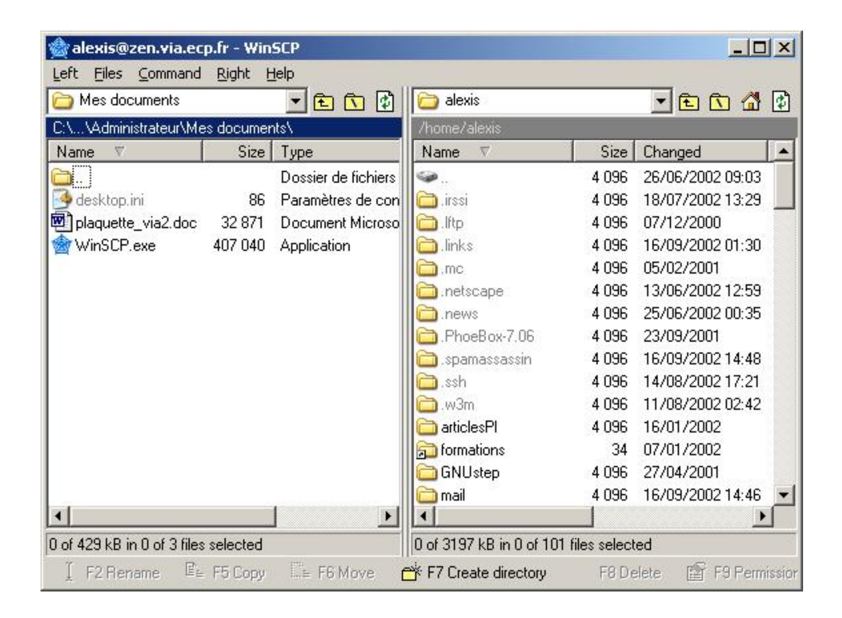


FIGURE G.4 – WinSCP

G.3 Accéder à ses partitions Ext2 ou Ext3 depuis Windows

Deux logiciels permettent de faire cela :

Explore2fs

Explore2fs est un programme GPL qui permet de lire ses partitions Linux de type Ext2 et Ext3 depuis Windows.

Attention

Pour se servir de ce programme sous Windows NT / 2000 / XP Pro, il faut avoir les privilèges d'administrateur.

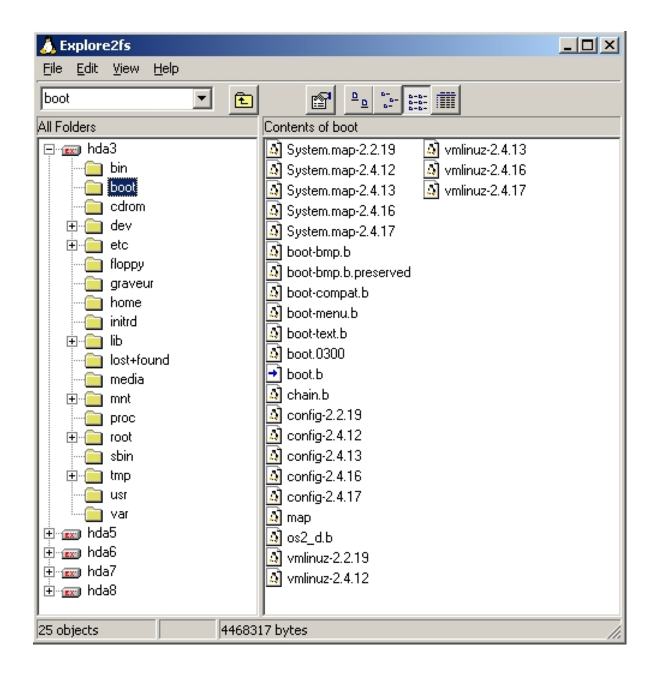


FIGURE G.5 – Explore2fs

Ext2 Installable File System

Ext2 Installable File System for Windows est un pilote pour le noyau de Windows qui permet d’accéder à ses partitions Linux de type Ext2 et Ext3 comme si c’était des partitions Windows. Les partitions sont alors accessible en lecture et en écriture depuis l’explorateur Windows et depuis n’importe quelle application Windows.

Annexe H

Monter un proxy-ARP

Note
Annexe écrite à partir d'une première version de Robert Cheramy.

Important
Cette partie requiert des connaissances de base en réseau Ethernet et IP. Lien vers une formation VIA à ce sujet.

H.1 L'idée

Le proxy-ARP rejoint le principe du bridge (expliqué dans l'annexe précédente 39) dans le sens où il permet de connecter plusieurs machines au réseau avec une machine centrale sous Linux. Par contre, contrairement au bridge qui agit au niveau Ethernet (i.e. layer 2), le proxy-ARP agit au niveau IP (i.e. layer 3). Cette annexe va donc vous apprendre à faire des tables de routage sous Linux !

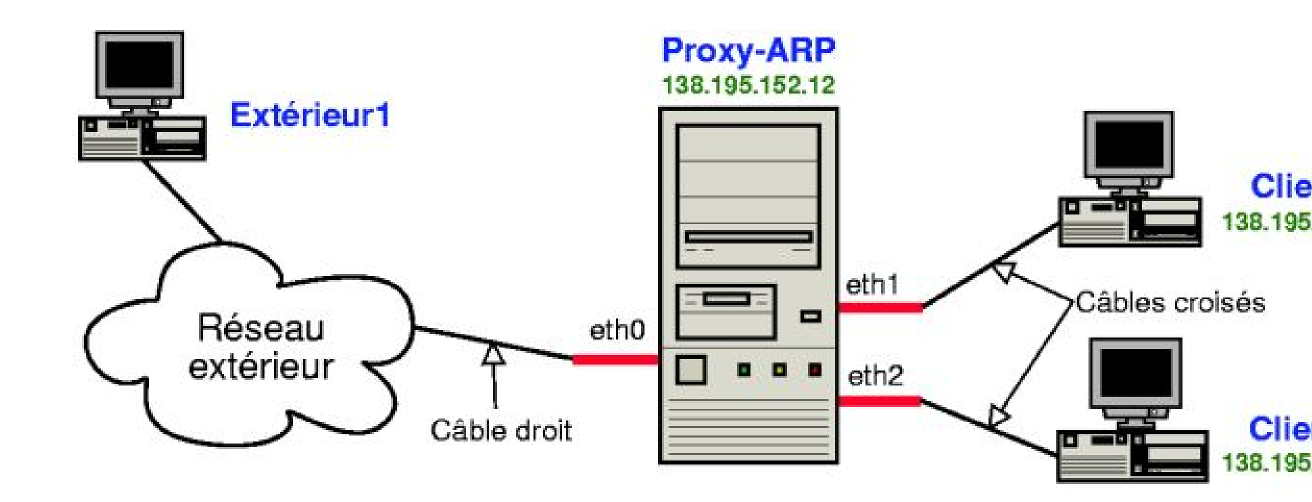


FIGURE H.1 – Schéma d'un proxy-ARP

H.2 Le principe de fonctionnement

La théorie

Le proxy-ARP marche un peu comme un routeur :

- points communs : il possède une table de routage et modifie les headers du niveau 2 en regardant les headers du niveau 3 ;
- différences : les clients qui sont derrière le proxy-ARP sont configurés normalement, comme si le proxy-ARP n'existait pas.

En pratique

- Communication de la machine *extérieur1* vers la machine *client1* :
  1. La machine *extérieur1* émet une requête ARP :

     ```
     "Qui est client1 ?" [ARP who-has client1]
     ```

  2. Le *proxy-arp* répond à la place de *client1* :

     ```
     "Je suis client1, j'attends tes paquets" [ARP ↩
         client1 is-at MAC_de_proxy-arp].
     ```

  3. Désormais, la machine *extérieur1* va transmettre tous ses paquets à destination de *client1* à *proxy-arp*. *Proxy-arp* se charge ensuite de les retransmettre à *client1* en mettant sa MAC comme MAC source.
- Communication de *client1* vers *extérieur1* :
  1. La machine *client1* émet une requête ARP :

     ```
     "Qui est extérieur1 ?" [ARP who-has extérieur1]
     ```

  2. Le *proxy-arp* répond à la place d' *extérieur1* :

```
"Je suis extérieur1, j'attends tes paquets" [ARP ↩
    extérieur1 is-at MAC_de_proxy-arp].
```

3. Désormais, la machine *client1* va transmettre tous ses paquets à destination d' *extérieur1* à *proxy-arp*. *Proxy-arp* se charge ensuite de les retransmettre à *extérieur1* en mettant sa MAC comme MAC source.

H.3 Montage

Préliminaires

Tout d'abord, la machine qui sert de proxy-ARP doit avoir plusieurs cartes réseau (autant que de machines derrière le proxy-ARP plus une carte réseau à connecter vers le réseau extérieur). Les modules correspondant à ces multiples cartes réseau doivent être compilés et installés. Les alias faisant la correspondance entre les interfaces réseau et les noms des modules à charger doivent être écrits dans un fichier du type `/etc/modprobe.d/reseau` contenant :

```
alias eth0 nom_du_module_de_la_carte_réseau_n\textdegree{}1
alias eth1 nom_du_module_de_la_carte_réseau_n\textdegree{}2
alias eth2 nom_du_module_de_la_carte_réseau_n\textdegree{}3
```

N'oubliez pas d'exécuter la commande **update-modules** après toute modification d'un fichier dans le répertoire `/etc/modprobe.d/`.

Vérifiez que toutes vos cartes réseau sont bien reconnues au démarrage.

Configuration du réseau du proxy-ARP

Pour plus de précisions concernant ce qui suit, je vous invite à consulter **man interfaces** et **man route**.

Nous allons maintenant modifier le fichier de configuration des interfaces réseau `/etc/network/interfaces` en utilisant mon fichier de configuration d'exemple et en le personnalisant :

```
# mv /etc/network/interfaces /etc/network/interfaces.old
# cp ~/fichiers-config/interfaces-proxy-arp /etc/network/ ↩
    interfaces
```

ou :

```
% wget http://formation-debian.via.ecp.fr/fichiers-config/ ↩
    interfaces-proxy-arp
# mv /etc/network/interfaces /etc/network/interfaces.old
# mv interfaces-proxy-arp /etc/network/interfaces
```

Personnalisez le nouveau fichier `/etc/network/interfaces` ; les lignes de commentaire doivent vous permettre de comprendre chaque paramètre :

```
# /etc/network/interfaces
# Fichier de configuration d'exemple des interfaces réseau
# pour faire un Proxy-ARP
# Formation Debian GNU/Linux par Alexis de Lattre
# http://formation-debian.via.ecp.fr/
```

```
# Plus d'informations dans "man interfaces" et "man route"


# L'interface de loopback
auto lo
iface lo inet loopback
    # Activation du "forwarding IP" et du "proxy-arp" au  ↩
        niveau du noyau :
    up echo "1" > /proc/sys/net/ipv4/ip_forward
    up echo "1" > /proc/sys/net/ipv4/conf/all/proxy_arp

# Configuration de l'interface eth0, connectée au réseau  ↩
    extérieur
auto eth0
iface eth0 inet static
    # Adresse IP du proxy-arp :
    address 192.168.0.12
    # Masque de sous-réseau du réseau extérieur :
    netmask 255.255.255.0
    # Adresse de la passerelle du réseau extérieur :
    gateway 192.168.0.1

# Configuration de l'interface eth1, connectée à client1
auto eth1
iface eth1 inet static
    # Adresse IP du proxy-arp :
    address 192.168.0.12
    # Masque du sous-réseau du réseau extérieur :
    netmask 255.255.255.0
    # Route qui dit que client1 est derrière eth1 :
    up route add 192.168.0.42 dev eth1
    # Suppression d'une route ajoutée à tort par la ligne  ↩
        précédente
    # 192.168.0.0 = adresse du réseau extérieur
    # 255.255.255.0 = masque de sous-réseau du réseau  ↩
        extérieur
    up route del -net 192.168.0.0 netmask 255.255.255.0 dev  ↩
        eth1

# Configuration de l'interface eth2, connectée à client2
auto eth2
iface eth2 inet static
    # Adresse IP du proxy-arp :
    address 192.168.0.12
    # Masque du sous-réseau du réseau extérieur :
    netmask 255.255.255.0
    # Route qui dit que client2 est derrière eth2 :
    up route add 192.168.0.43 dev eth2
    # Suppression d'une route ajoutée à tort par la ligne  ↩
        précédente
    # 192.168.0.0 = adresse du réseau extérieur
    # 255.255.255.0 = masque de sous-réseau du réseau  ↩
        extérieur
```

```
    up route del -net 192.168.0.0 netmask 255.255.255.0 dev  ↩
        eth2
```

Relancez la configuration des interfaces réseau :

```
# /etc/init.d/networking restart
```

Vérifiez que les changements ont bien été pris en compte :

```
% ifconfig
```

Vérifiez que la table de routage est bonne :

```
% route -n
```

Dans l'exemple de ce chapitre, la table de routage est la suivante :

```
Table de routage IP du noyau
Destination     Passerelle      Genmask         Indic Metric  ↩
    Ref    Use Iface
192.168.0.42  0.0.0.0         255.255.255.255 UH    0      0  ↩
           0 eth1
192.168.0.43  0.0.0.0         255.255.255.255 UH    0      0  ↩
           0 eth2
192.168.0.0   0.0.0.0         255.255.255.0   U     0      0  ↩
           0 eth0
0.0.0.0         192.168.0.1   0.0.0.0         UG    0      0  ↩
           0 eth0
```

Configuration du réseau des clients

Configurer un client Linux

La configuration des clients est strictement identique à la configuration qu'ils auraient s'ils n'étaient pas derrière le proxy-ARP. Par contre, il faut définir leur IP en dur, pas par DHCP, car le broadcast est bloqué par le proxy-ARP.

Si le client est aussi une Debian, éditez le fichier `/etc/network/interfaces` :

```
# /etc/network/interfaces de client1

auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
    # Adresse IP de client1 :
    address 192.168.0.42
    # Masque de sous-réseau du réseau extérieur :
    netmask 255.255.255.0
    # Adresse de la passerelle du réseau extérieur :
    gateway 192.168.0.1
```

Pour que le système tienne compte des modifications :

```
# /etc/init.d/networking restart
Reconfiguring network interfaces: done.
```

Configurer un client Windows

La configuration Windows est semblable, si vous avez compris, ça devrait aller.

Faire du DHCP relay

Le proxy-ARP bloque le broadcast des clients ; donc si ces derniers font une requête DHCP, elle n'atteindra pas le réseau extérieur. Pour pallier à ce problème et faire en sorte que les clients puissent être configurés par DHCP, il faut installer un *relai DHCP* sur le proxy-ARP.

Pour cela, installez le paquet suivant :

```
# aptitude install dhcp3-relay
```

Lors de la configuration du paquet, il vous demande :

1. *What DHCP servers should the DHCP relay forward requests to ?* Entrez l'adresse IP du serveur DHCP du réseau extérieur.
2. *On what network interfaces should the DHCP server listen ?* Si toutes les interfaces sont utilisées pour faire le proxy-ARP, comme c'est le cas dans cet exemple, laissez le champ vide et validez.

Le fichier de configuration `/etc/default/dhcp3-relay` est alors généré, et le démon *dhcrelay3* lancé. Si vous avez besoin d'arrêter ou de relancer le démon, utilisez le script `/etc/init.d/dhcp3-relay` avec le bon argument.

Vous pouvez maintenant configurer le réseau de *client1* et *client2* par DHCP.

Astuce

Pensez a rajouter les IP de *client1*, *client2* et *proxy-arp* dans les `/etc/hosts` des trois machines ; c'est plus pratique pour travailler quand on est coupé du réseau...

Annexe I

Faire marcher une connexion sans fil

I.1 Se renseigner sur sa carte wifi

Il va falloir compiler les pilotes de votre carte wifi. Pour cela, il faut d'abord voir de quel type de carte il s'agit. Pour cela, lancez la commande :

```
$ lspci
```

Si vous avec un portable avec un processeur Centrino, par exemple, il s'agit sûrement d'une *Intel pro wireless 2200.*

I.2 Compiler le pilote noyau

Dans tous les cas, il faut activer le support du protocole IEEE 802.11 (wifi, quoi) :

```
Networking --> Generic IEEE 802.11 Networking Stack
```

Votre pilote est disponible dans les sources du noyau

Plusieurs pilotes wifi sont disponibles dans le noyau, dans la section :

```
Device Driver --> Network device support --> Wireless LAN ( ↩
    non-hamradio)
```

Choisissez le pilote adapté à votre carte. Si vous comptez utiliser une réseau sécurisé WEP ou WPA, dans la section :

```
Cryptographic options
```

choisissez également :

```
<M>   ARC4 cipher algorithm
<M>   Michael MIC keyed digest algorithm
<M>   AES cipher algorithms
```

Compilez ensuite votre nouveau noyau, et démarrez dessus.

Votre pilote n'est pas dans les sources du noyau

Si votre pilote n'est pas dans les sources du noyau, il existe sans doute un pilote libre, à compiler en externe, indépendamment du noyau. Les carte *Athereos* disposent ainsi d'un pilote *MADWifi*. Cherchez le pilote adapté à votre carte sur Internet, décompressez-le, et suivez les instructions données dans les README et INSTALL...

Chargez ensuite votre nouveau pilote. Pour une carte Athereos, utilisez la commande :

```
# modprobe madwifi
```

I.3 Installer les outils de connexion

Installez les outils de connexion sans fil :

```
# aptitude install wireless-tools
```

Si vous comptez utiliser une connexion sécurisée par WPA, installez également les outils correspondants :

```
# aptitude install wpasupplicant
```

I.4 Configurer sa connexion

Vous disposez maintenant d'une interface réseau supplémentaire, ce que vous pouvez vérifier avec :

```
$ ifconfig -a
```

Les outils de connexion sans fil fournissent une commande semblable, pour les réglages spécifiques aux connexions sans fil :

```
$ iwconfig
```

Vous pouvez également lister les réseaux sans fil qui vous entourent :

```
$ iwlist [interface] scan
```

Connexion non sécurisée

Pour vous connecter, il suffit de s'associer à un réseau. Ainsi, pour vous connecter au réseau « maison », en supposant que votre carte réseau correspond à l'interface *eth1* :

```
# iwconfig eth1 essid maison
```

Connexion cryptée WEP

Pour vous connecter à un réseau crypté avec WEP, il suffit de préciser en plus la clef WEP :

```
# iwconfig eth1 essid VIA key XXXXXXXX
```

Configuration IP

Vous êtes maintenant connecté à un réseau sans fil, ce que vous pouvez vérifier par la commande :

```
$ iwconfig
```

Cependant, vous êtes simplement branché sur un réseau, comme vous le seriez par un câble. Il faut donc maintenant régler les paramètres IP de la connexion. Si vous avec un serveur DHCP :

```
# dhclient eth1
```

Comme pour une interface filaire, vous pouvez aussi préciser ces paramètres manuellement, connaissant les réglages du réseau. Ici, pour un réseau privé typique :

```
# ifconfig eth1 192.168.0.42 netmask 255.255.255.0
# route add default gw 192.168.0.1 eth1
```

Configurer définitivement sa connexion

Vous pouvez rajouter une section dans le fichier `/etc/network/interfaces` pour votre connexion sans fil. Par exemple :

```
iface eth1 inet static
  address 192.168.0.42
  netmask 255.255.255.0
  broadcast 192.168.0.255
  wireless-essid maison
  wireless-key XXXXXXXX
```

Annexe J

Trucs et Astuces

J.1 Allumer le verrouillage numérique

Les utilisateurs d'un ordinateur fixe sont souvent habitués à utiliser le pavé numérique... mais le NumLock n'est jamais allumé par défaut sous Linux !

en console

Décommentez les lignes suivantes dans le fichier `/etc/zsh/zlogin` :

```
# Pour les ordinateurs avec un pavé numérique...
# Active le pavé numérique quand on se loggue en console
case "`tty`" in /dev/tty[1-6]*)-
    setleds +num
esac
```

sous X

Installez le paquet requis :

```
# aptitude install numlockx
```

Lors de l'installation du paquet, il vous demande *Enable NumLock automatically ?* : répondez *Oui*.

Si vous utilisez GDM, il faut ajouter le script de lancement de *numlockx* au script de lancement de GDM. Pour cela, éditez le fichier `/etc/gdm/PreSession/Default` et importez le contenu du fichier `/etc/X11/Xsession.d/55numlockx` après la première ligne qui commence par *PATH=* (commande **:r nom_du_fichier** pour importer un fichier sous *vim*), puis redémarrez GDM.

J.2 Brancher un périphérique IDE à chaud

Si vous avez un portable muni d'une media-bay et de périphériques IDE amovibles, un utilitaire très pratique permet l'ajout et le retrait à chaud de ces périphériques.

Attention

Les lecteurs de disquettes ne sont pas gérés par cet utilitaire.

Installez le paquet *hotswap-text* :

```
# aptitude install hotswap-text
```

L'utilisation est très simple ; il suffit de taper le nom du programme en root :

```
# hotswap
```

et de se laisser guider par le programme comme illustré ci-dessous.

– Insertion d'un lecteur DVD :

```
0:43 root@alpy ~# hotswap
I/O warning : failed to load external entity "/etc/ ↩
    hotswaprc"
hotswap 0.4.0
Copyright 2001 Tim Stadelmann
This is free software, licensed under the conditions of  ↩
    the
GNU General Public License version 2, or (at your option ↩
    ), any later
version.

Il n'y a actuellement aucun périphérique IDE configured. ↩
     (Lecteurs de
disquettes, batteries, et 'modules de voyage' ne sont  ↩
    pas gérés par cet
utilitaire. Si vous voulez échanger un tel module, vous  ↩
    devriez le faire
maintenant.)

Voulez-vous insérer un périphérique IDE dans la baie ?y
Insérez le nouveau périphérique dans la baie et pressez  ↩
    la touche Entrée.


Le périphérique IDE suivent a été correctement configuré ↩
    :
HL-DT-STDVD-ROM GDR8081N
```

– Retrait d'un lecteur DVD :

```
0:44 root@alpy ~# hotswap
I/O warning : failed to load external entity "/etc/ ↩
    hotswaprc"
hotswap 0.4.0
Copyright 2001 Tim Stadelmann
This is free software, licensed under the conditions of  ↩
    the
GNU General Public License version 2, or (at your option ↩
    ), any later
version.
```

```
Le périphérique IDE suivant est actuellement configuré:
HL-DT-STDVD-ROM GDR8081N
Voulez-vous retirer ce périphérique ?y
Vous pouvez maintenant retirer le périphérique de la ↩
    baie.

Voulez-vous insérer un périphérique IDE dans la baie ?n
Annulation
zsh: exit 1     hotswap
```

J.3 Faire du SSH à travers un firewall

Cas typique : vous êtes dans un entreprise et vous voulez vous connecter par SSH vers un serveur à l'extérieur du réseau de l'entreprise... mais il y a un firewall entre le réseau interne et Internet !

AVERTISSEMENT
L'utilisation des techniques décrites ci-dessous est peut-être interdite par l'entreprise.

Il y a trois possibilités, sachant que seule la dernière est potentiellement réalisable si vous ne pouvez pas modifier la configuration du serveur.

Changer de port

Si le firewall a un port complètement ouvert (le port 80 par exemple), vous pouvez modifier la configuration du serveur pour que son démon SSH écoute sur le port 80 en plus du port 22 (attention, il ne doit pas y avoir de serveur Web qui écoute déjà sur le port 80). Pour cela, modifiez le fichier de configuration `/etc/ssh/sshd_config` pour qu'il contienne les deux lignes suivantes :

```
Port 22
Port 80
```

Puis relancez le serveur SSH :

```
# /etc/init.d/ssh reload
Reloading OpenBSD Secure Shell server's configuration.
```

Vous pouvez alors lancer votre client SSH en lui précisant d'utiliser le port 80 :

```
% ssh -p 80 login@nom_DNS_du_serveur
```

Monter un tunnel HTTP

Si le firewall surveille le port 80 et vérifie que seuls des packets HTTP passent par ce port, alors vous pouvez essayer d'utiliser le programme **httptunnel**. Ce programme doit tourner sur le client *et* le serveur. Il encapsule dans une connexion HTTP n'importe quel type de connexion (pas forcément une connexion SSH). Il permet aussi de faire passer la connexion par un Proxy. Installez le paquet sur le client et sur le serveur :

```
# aptitude install httptunnel
```

Lancez le serveur *httptunnel* sur le serveur :

```
% hts -F localhost:22 8888
```

Puis lancez le client *httptunnel* sur le client :

```
% htc -F 2222 nom_DNS_du_serveur:8888
```

ou, si vous devez passer par un Proxy *proxy.exemple.org* qui écoute sur le port *8080* :

```
% htc -F 2222 -P proxy.exemple.org:8080 nom_DNS_du_serveur ↩
   :8888
```

Vous pouvez alors lancer votre client SSH :

```
% ssh -p 2222 login@localhost
```

Utiliser un proxy HTTPS

Récupérez le script `ssh-https-tunnel` :

```
% wget http://zwitterion.org/software/ssh-https-tunnel/ssh- ↩
   https-tunnel
# mv ssh-https-tunnel /usr/local/bin/
```

Editez le fichier `/usr/local/bin/ssh-https-tunnel` contenant le programme et rentrez le nom DNS et le port du proxy du réseau interne dans les variables prévues à cet effet :

```
# Proxy details
my $proxy      = "proxy.exemple.org";
my $proxy_port = 8080;
```

Créez (s'il n'existe pas déjà) un fichier `~/.ssh/config` contenant :

```
host nom_DNS_du_serveur
   ProxyCommand /usr/local/bin/ssh-https-tunnel %h %p
   Port 22
```

Puis essayez de vous connecter au serveur comme vous le faites normalement :

```
% ssh login@nom_DNS_du_serveur
```

Si cela ne marche pas, essayez de faire écouter le démon SSH du serveur sur le port 443 (port des connexions HTTP sécurisées), comme expliqué dans la première possibilité, et modifiez le fichier `~/.ssh/config` en remplaçant **22** par **443**, puis réessayez de vous connecter.

Annexe K

GNU General Public License

K.1 Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software - to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps:

1. copyright the software, and
2. offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have

is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

K.2 TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

Section 0

This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a « work based on the Program » means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term « modification ».) Each licensee is addressed as « you ».

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

Section 1

You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

Section 2

You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of K.2 above, provided that you also meet all of these conditions:

1. You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
2. You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

3. If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License.

   Exception:
   If the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

Section 3

You may copy and distribute the Program (or a work based on it, under K.2 in object code or executable form under the terms of K.2 and K.2 above provided that you also do one of the following:

1. Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
2. Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
3. Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special

exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

Section 4

You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

Section 5

You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

Section 6

Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

Section 7

If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

Section 8

If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

Section 9

The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

Section 10

If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY Section 11

BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH

YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

Section 12

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

K.3 How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.> Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called

something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989 Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

Septième partie

Glossaire

FSF (free software foundation)

Une association, créée en 1985 par Richard Stallman, pour aider au développement du projet GNU et promouvoir le logiciel libre.

GPL (general public licence)

Licence logicielle écrite pour donner un cadre juridique aux logiciels du projet GNU

Cette licence vise à garantir les quatre libertés fondamentales du logiciel libre, et à empêcher leur restriction, en utilisant la notion de *copyleft* : l'utilisation et la modification des logiciels sous GPL est librement permise, dans la mesure où ces mêmes libertés restent préservées. Il est donc impossible de refermer un logiciel sous GPL, ou de s'en servir comme base pour un logiciel propriétaire, c'est pourquoi on qualifie cette licence de « virale »

GNU (GNU's not UNIX)

Un projet de système d'exploitation composé exclusivement de logiciels libres. Lancé par Richard Stallman, ce système se veut compatible avec UNIX. Actuellement, le projet GNU a permis de développer de très nombreux outils de base du système d'exploitation, mais le noyau, GNU Hurd n'est pas encore pleinement fonctionnel : les outils GNU sont donc souvent utilisés avec le noyau Linux, formant un système GNU/Linux.

POSIX (portable operating system interface X)

Une famille de standard définis par l'IEEE pour standardiser les interfaces de programmation des systèmes Unix-like

Index